ONERA
THE FRENCH AEROSPACE LAB

Scientific Day 2012

Multidisciplinary Analysis Workflow with the FlowSimulator

Lars Reimer

With contributions of
Daniel Vollmer, Gunnar Einnarson, Stefan Görtz, Thomas Gerhold,
Ralf Heinrich, Norbert Kroll, Andreas Michler,
Markus Ritter, Jens Neumann (all DLR),
Lars-Uwe Hansen (Airbus)

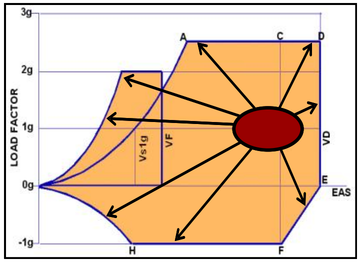
DLR

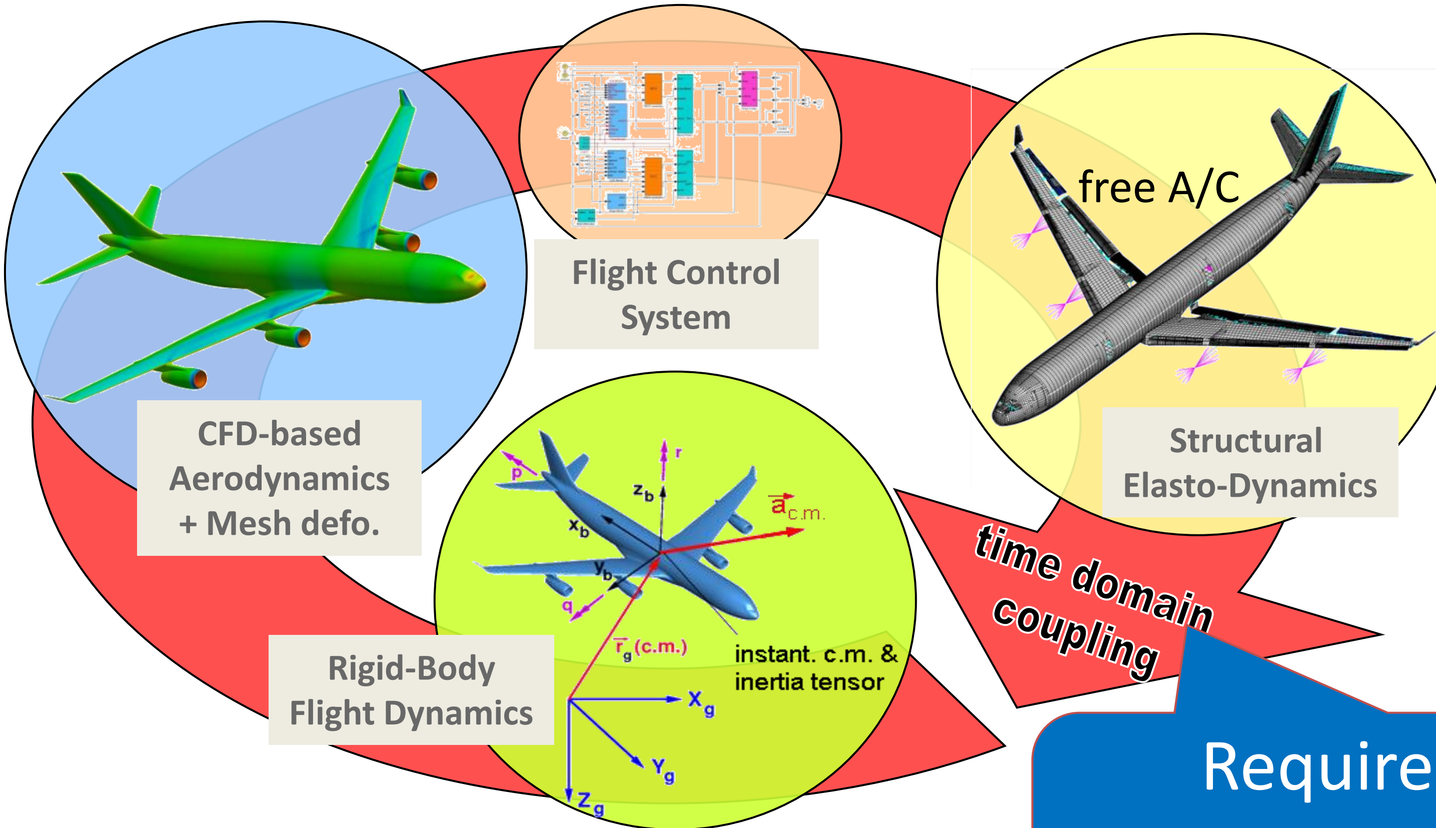
Outline

- General overview of FlowSimulator (objectives, concept, etc.)
- Aspects of trim simulations with FS
- Aspects of CFD-CSM coupled simulations with FS

focus on DLR process chains

Overall Objective: Accurate Analysis for Entire Flight Envelope





Requires Efficient Multidisciplinary HiFi Simulation Tools

DLR

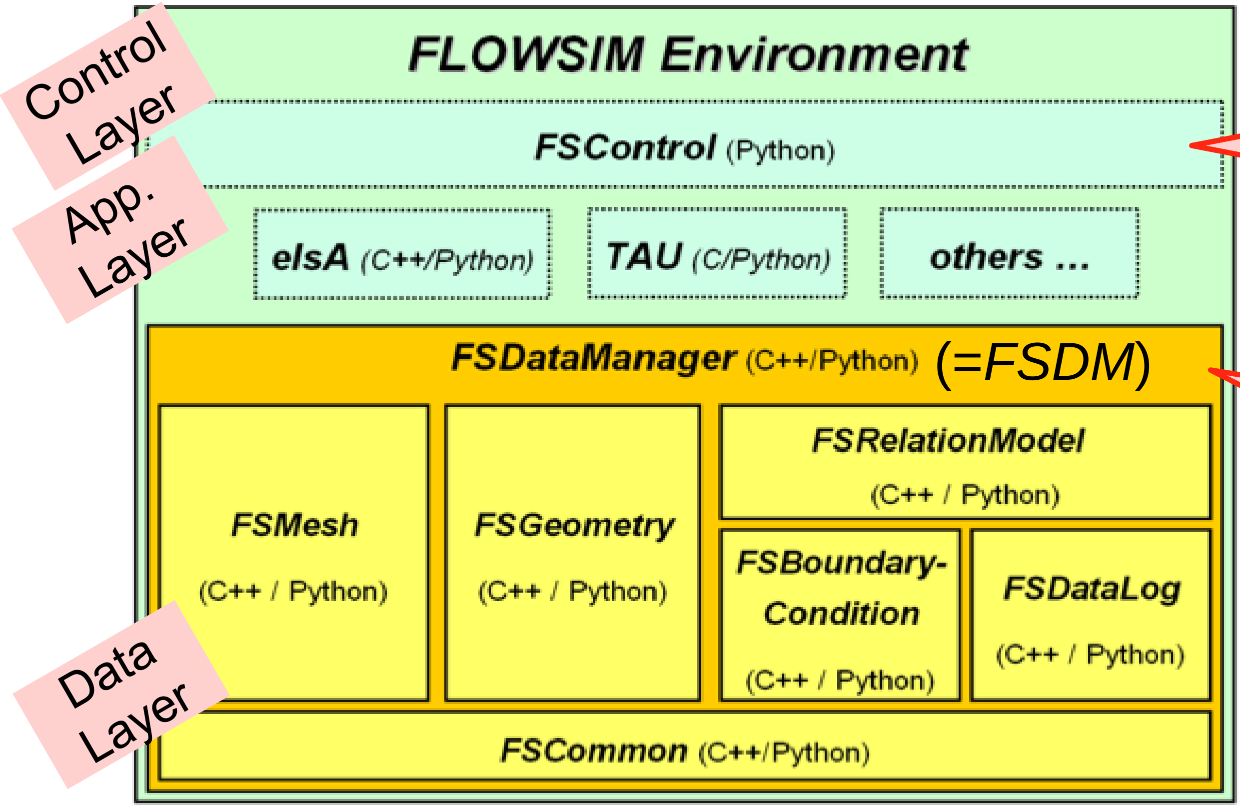
What is FlowSimulator and What Are Its Main Objectives ?

- Numerical tool box for **high-performance multi-disciplinary** simulations
- Designed for efficient massively-parallel **in-memory data exchange** between mono-disciplinary codes
- **Easy replacement** of simulation components

DLR

What is *FlowSimulator* Technically ?

- *FS* is a bundle of Python modules which work on a common data structure, i.e. the *FSDM*
- Python-based scripting layer enables rapid prototyping of tool chains



- devel. mainly by Airbus
- supposed to be main access layer for end-users (high-level classes)

- devel. by entire FS community
- open source
- provides parallel data management

GForge Server (http://dev.as.dlr.de/gf)

FORGE

Home | Meine Sachen | Benutzer | Suche | Projekte | Schnipsel

Check for list of existing projects

Home » Project Durchsuchen

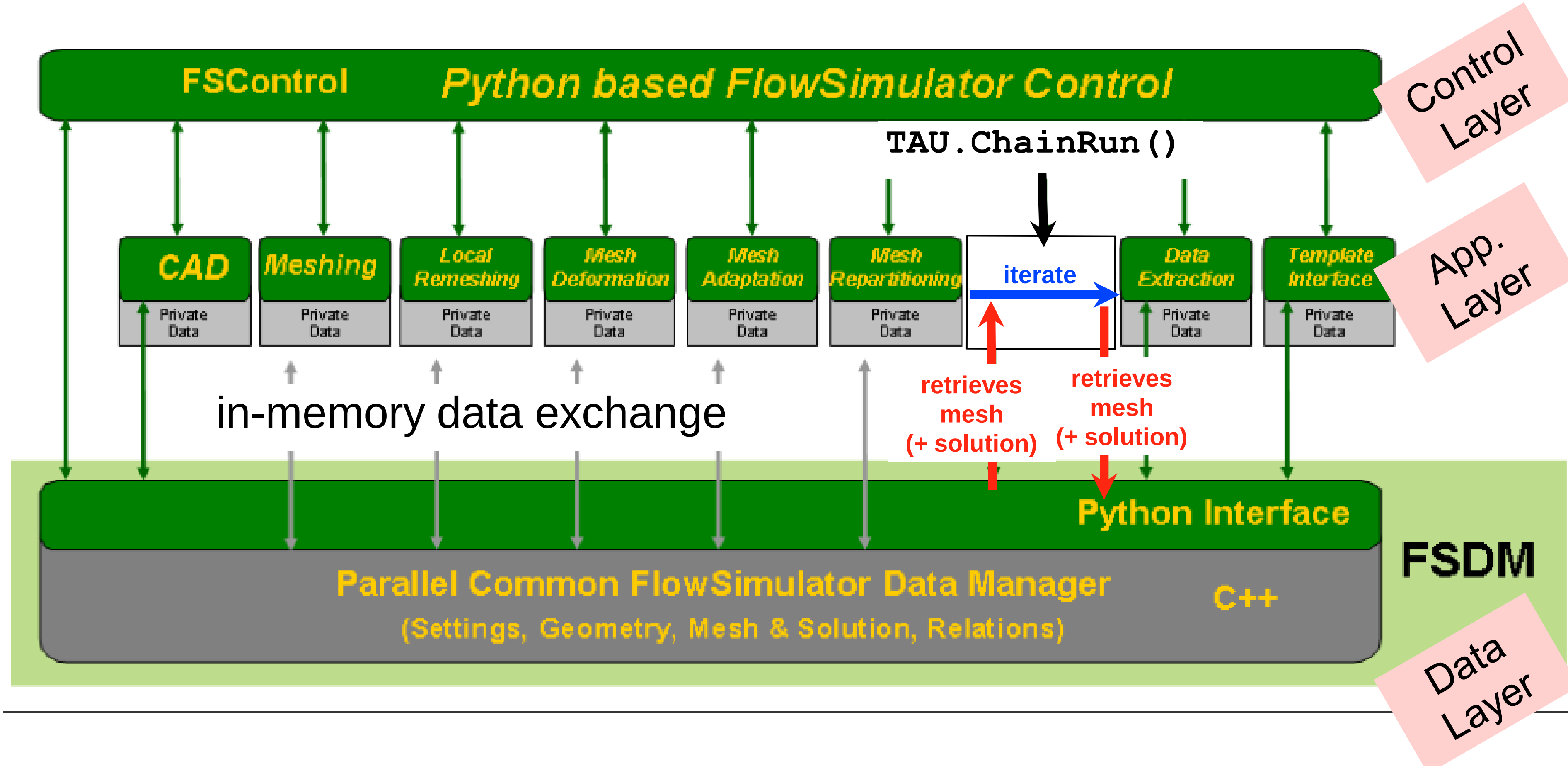
Vollständige Bezeichnung	Accountname (Kleinbuchstaben)	Beschreibung
TAU Solver (FS-Mod)	fstau	TAU is a highly sophisticated unstructured CFD solver for compressible flows.
SELENE	selene	Computation of flight-over-noise and acoustic optimization of flight procedures of helicopters (EC135, BO105)
FSTrim	fstrim	Trim module of FlowSimulator
FSPrimgrid	fsprimgrid	This is the in-memory converter for TauPrimgrid and FSMesh.
FSNumPyInterface	fsnumpy	This module gives you a way to treat FSArrays in Python as NumPy arrays.
FSMeshConverter	fsmeshconvert	This plugin allows you to convert structured meshes to unstructured ones.
FSForce	fsforce	FSForce - generic friction and pressure coefficient integration for structured and unstructured meshes
FSExtraConnectivity	fsextraconn	A FSDM module that discovers extra connectivity information.
FSelsA	fselsa	A FlowSimulator plugin providing bindings to elsA.
FSEADM	fseadm	Provide EADM tool suite bindings for FlowSimulator.
FSDM Zoltan Partitionier Integration	fszoltan	This FSDM extension (aka MeshOp) integrates the Zoltan PHG hypergraph partitioner.
FSDM Support	fsdmsupport	This project bundles some small modules that are optional requirements for FlowSimulator DataManager.
FSDM CGNS-IO	fscgnsio	Extension of the Import/Export filt
FSDemoData	fsdemodata	Example data for use with FSDM.
FSDeformation	fsdeformation	Deformation module for FSDM
FSDamasIO	fsdamasio	DAMAS in-/output for FlowSimulato
FSCouple	fscouple	A generalized coupling framework f
FSCatia interface	fscatia	Provides a tool to export CATIA ge ... DataManager.
fsbsurf	fsbsurf	A FlowSimulator module for BSurf P
FSAdvancedSplining	fsadvsplining	FlowSimulator-plugin dealing with sp

- Join *FlowSimulator* projects of your interest
- Access developer releases
- Follow development process (commit mess.) & bugfixes

DLR

FS' Design Dogma for Replaceability of Sim. Components

- NO horizontal data exchange between simulation components



Trim Simulations with FlowSimulator

General Longitudinal Trim Process

Initial CFD solution

While trim condition **not** fulfilled (here $C_L = C_{L,Target}$, $C_{My} = 0$):

LOOP

If Jacobian is to be updated:

<u>Computation of Jacobian:</u>
- Perturbation of control variable (here AoA, HTP)
- Transfer perturbed control variable into BCs/mesh
- Compute CFD sol. for perturbed state

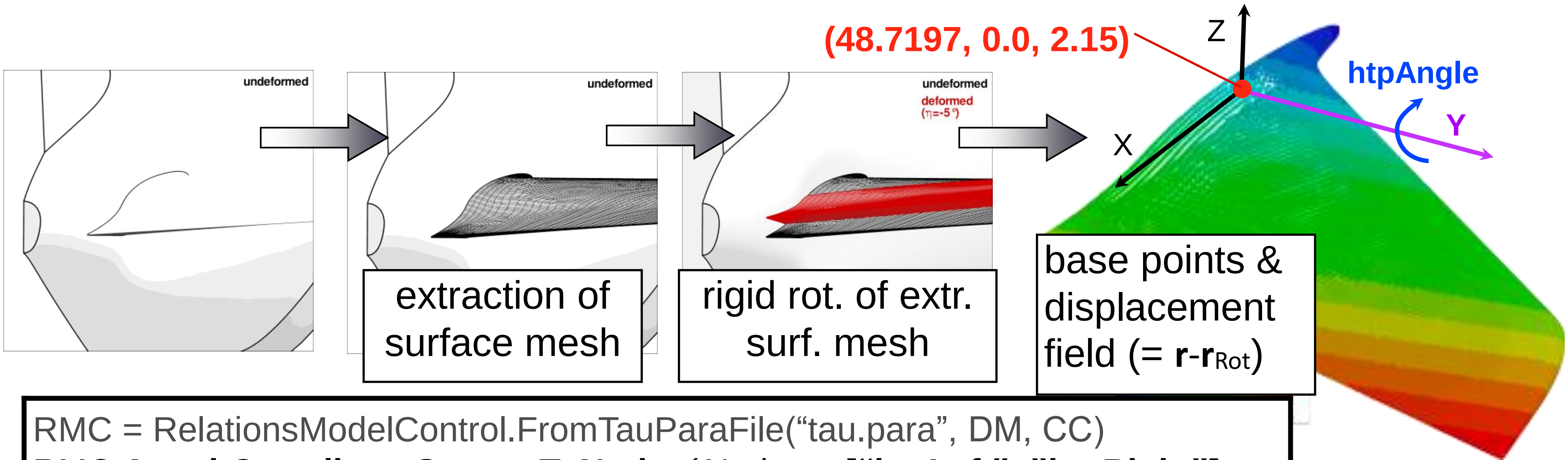
Compute Newton step (here for AoA & HTP)

Transfer altered control variable into BCs/mesh

Compute CFD solution

Challenge: Rotation of HTP while preserving the fuselage geometry

Computation of the Scattered Displacement Field for the HTP Rotation: FS' MeshTools (part of FSControl)




```
RMC = RelationsModelControl.FromTauParaFile("tau.para", DM, CC)
RMC.AttachCoordinateSystemToNodes( Nodes = ["htpLeft", "htpRight"],
                                   CommonNode = "HTP",
                                   Origin = [48.7197, 0.0, 2.15])
```

define rotation axis in relation model

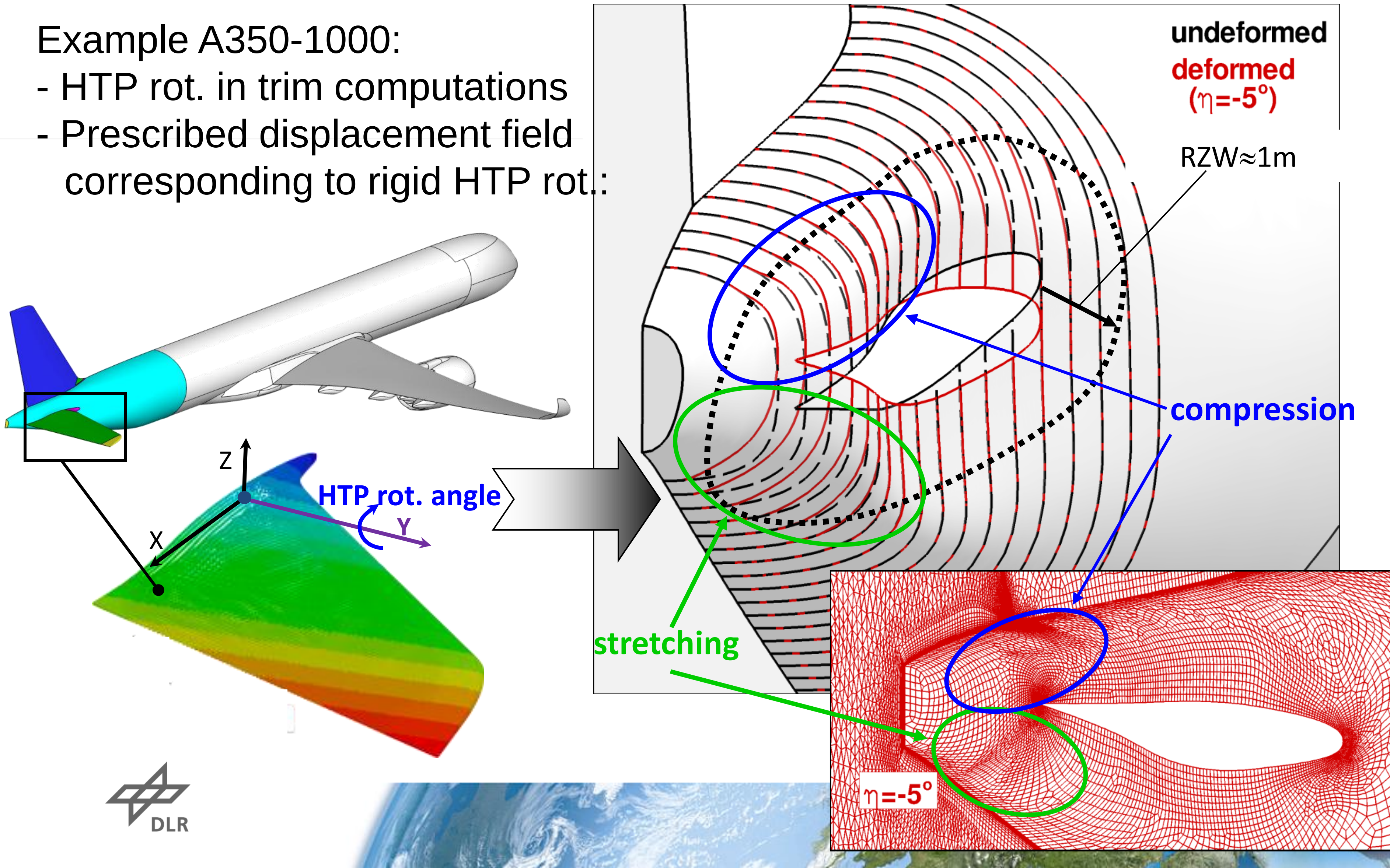
```
MeshTools = MeshTools(DM, RMC)
basePoints, displVecs = \
        MeshTools.GetDefoVectorsForRotation( MeshKey = "myMesh",
        ModelNodeName = "HTP",
        Axis = "Y",
        Angle = htpAngle)
```

apply rotation & return displ. field

DLR

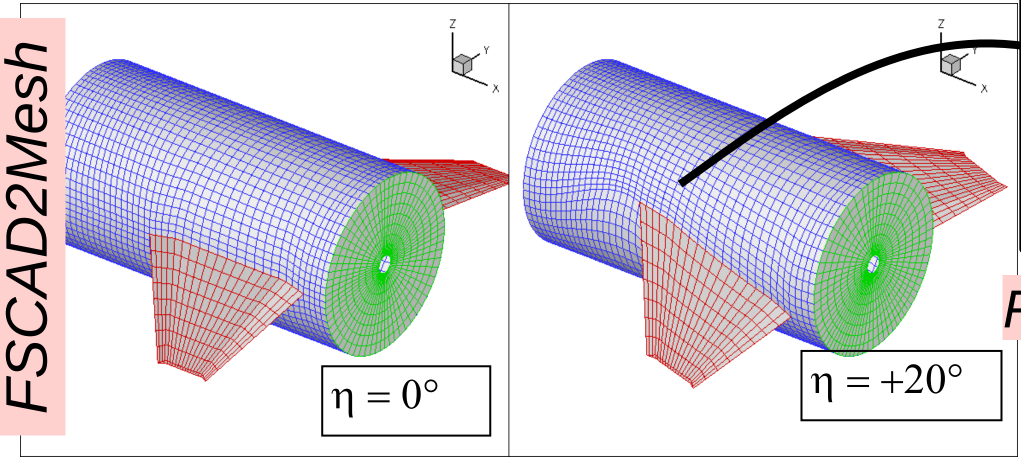
Problem with Mesh Deformation at HTP-Fuselage Intersection

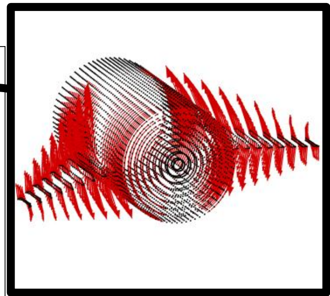
Example A350-1000:
- HTP rot. in trim computations
- Prescribed displacement field corresponding to rigid HTP rot.:



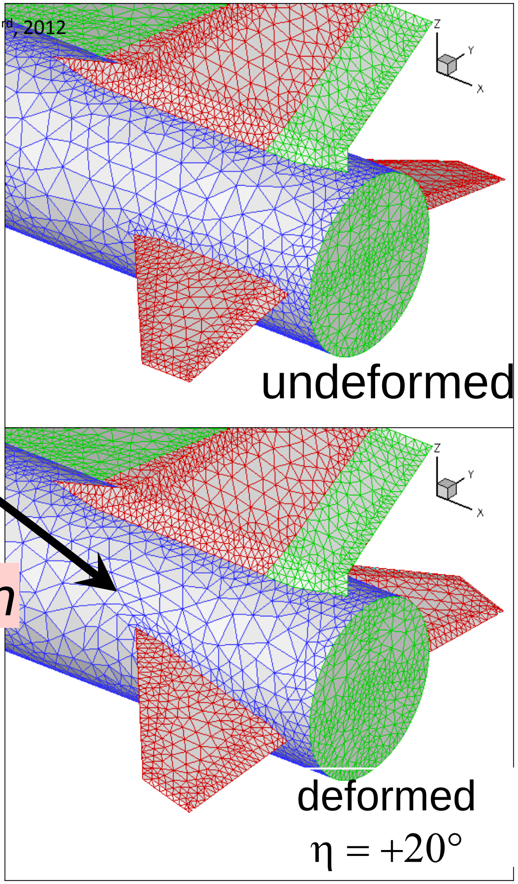
CAD-based Preparation of the Scattered Displacement Fields + Mesh Deformation: FSCAD2Mesh + FSDefoFields

FSCAD2Mesh





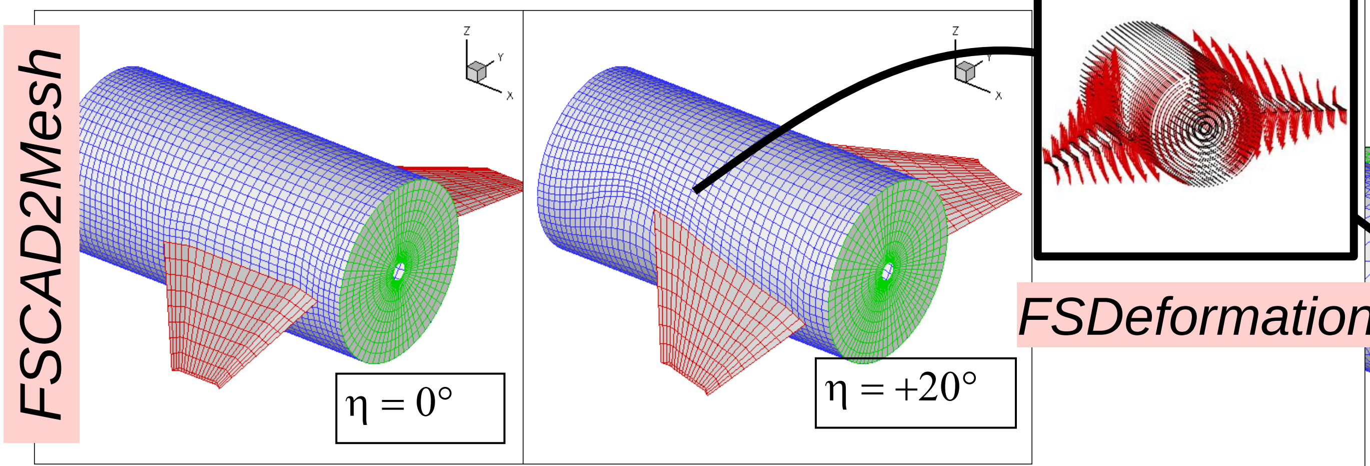
FSDeformation

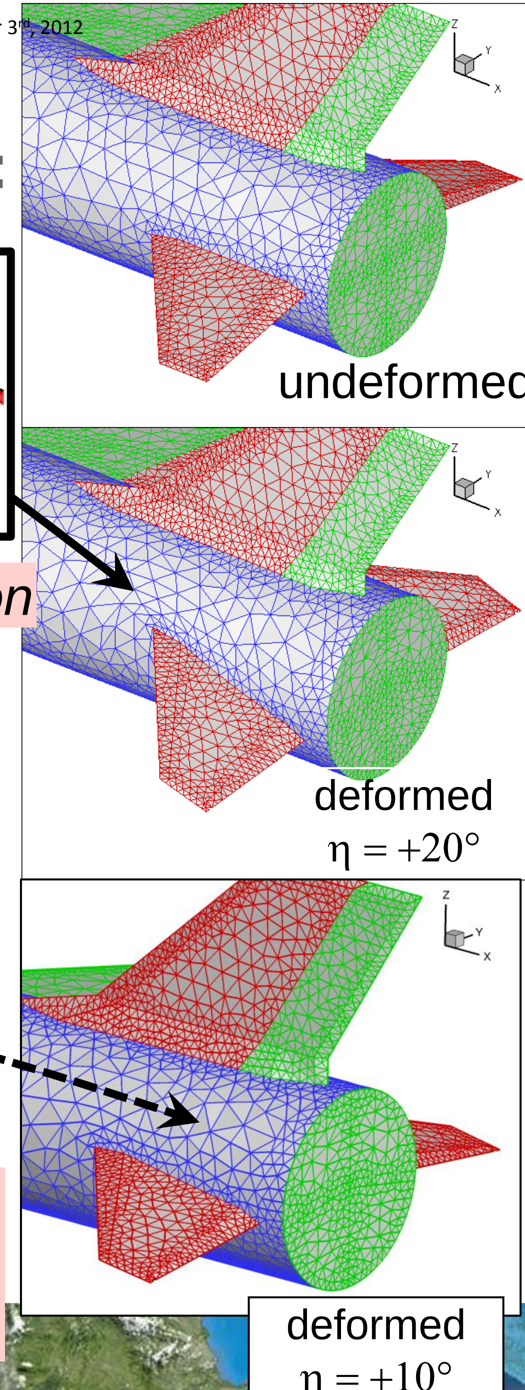


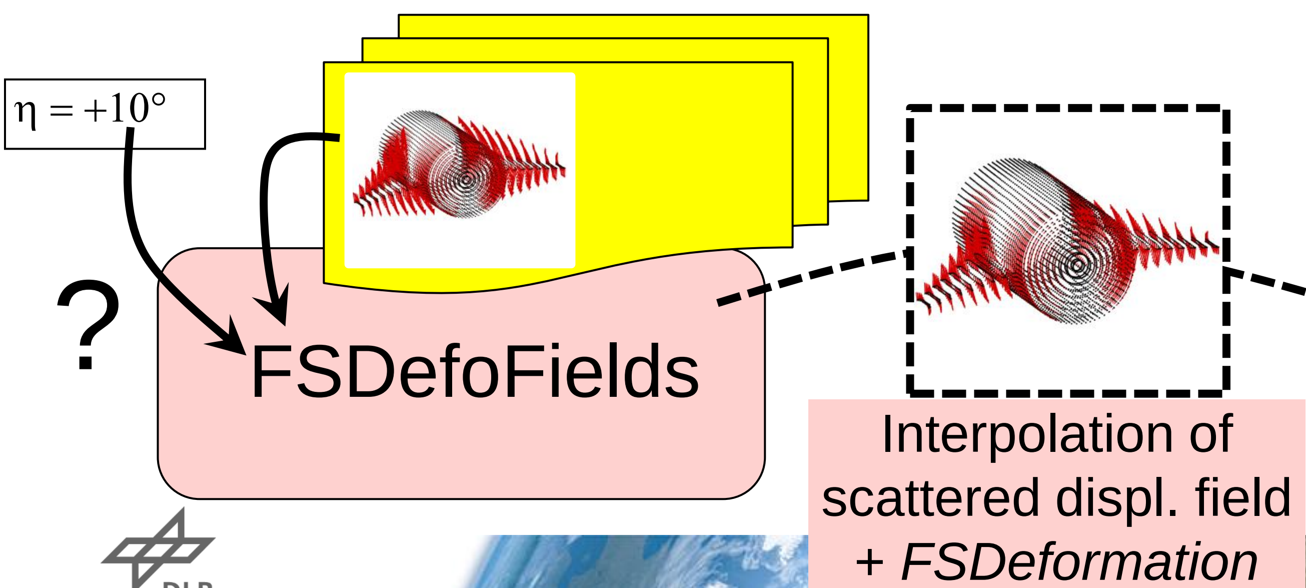
$\eta = +10°$

?

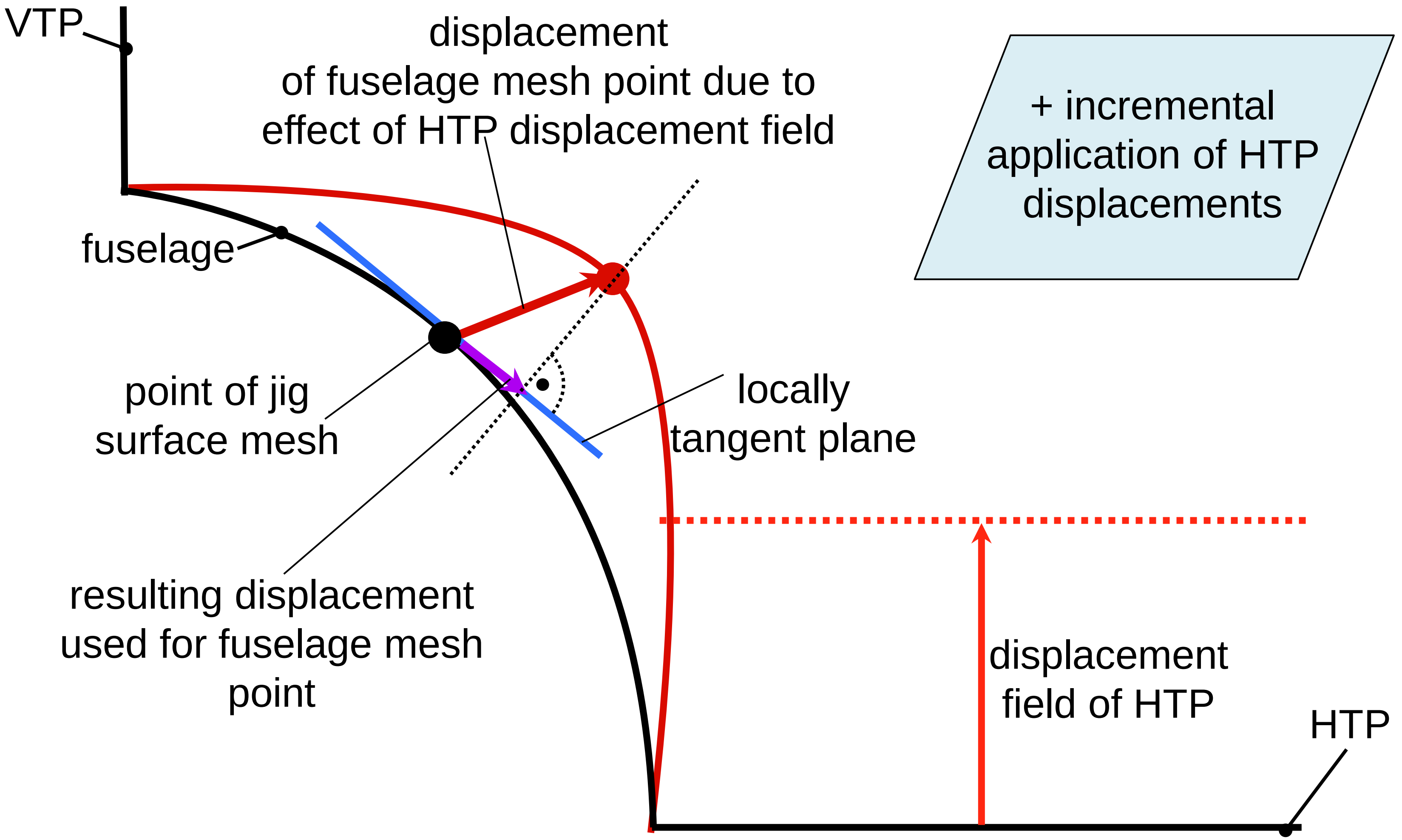
CAD-based Preparation of the Scattered Displacement Fields + Mesh Deformation: FSCAD2Mesh + FSDefoFields





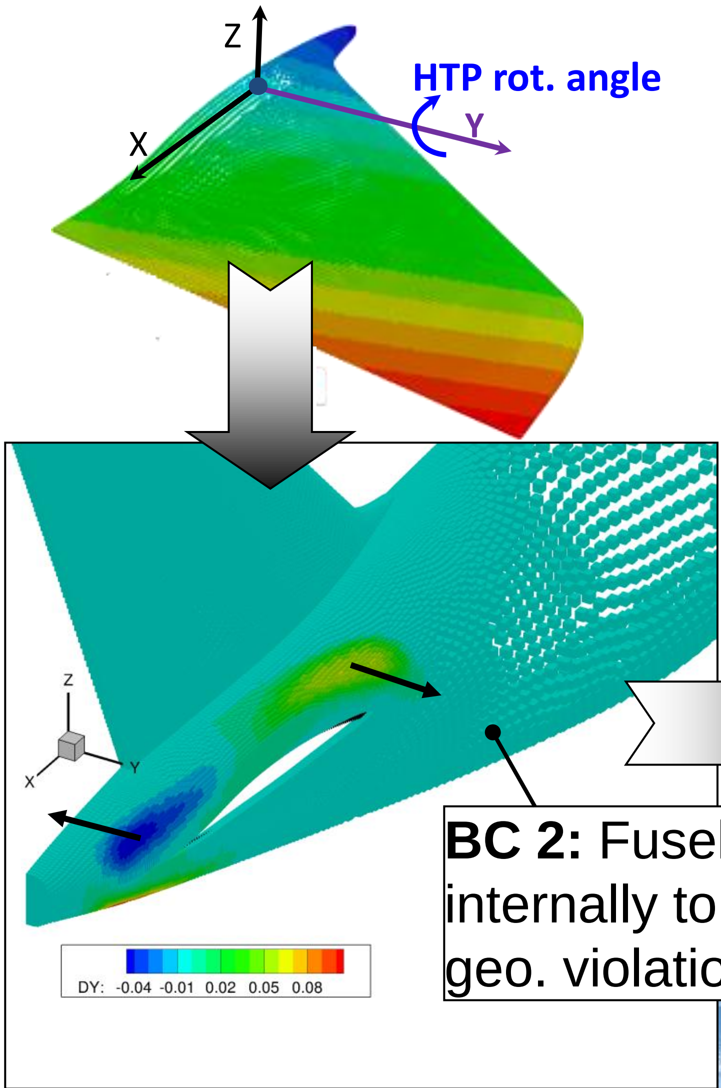


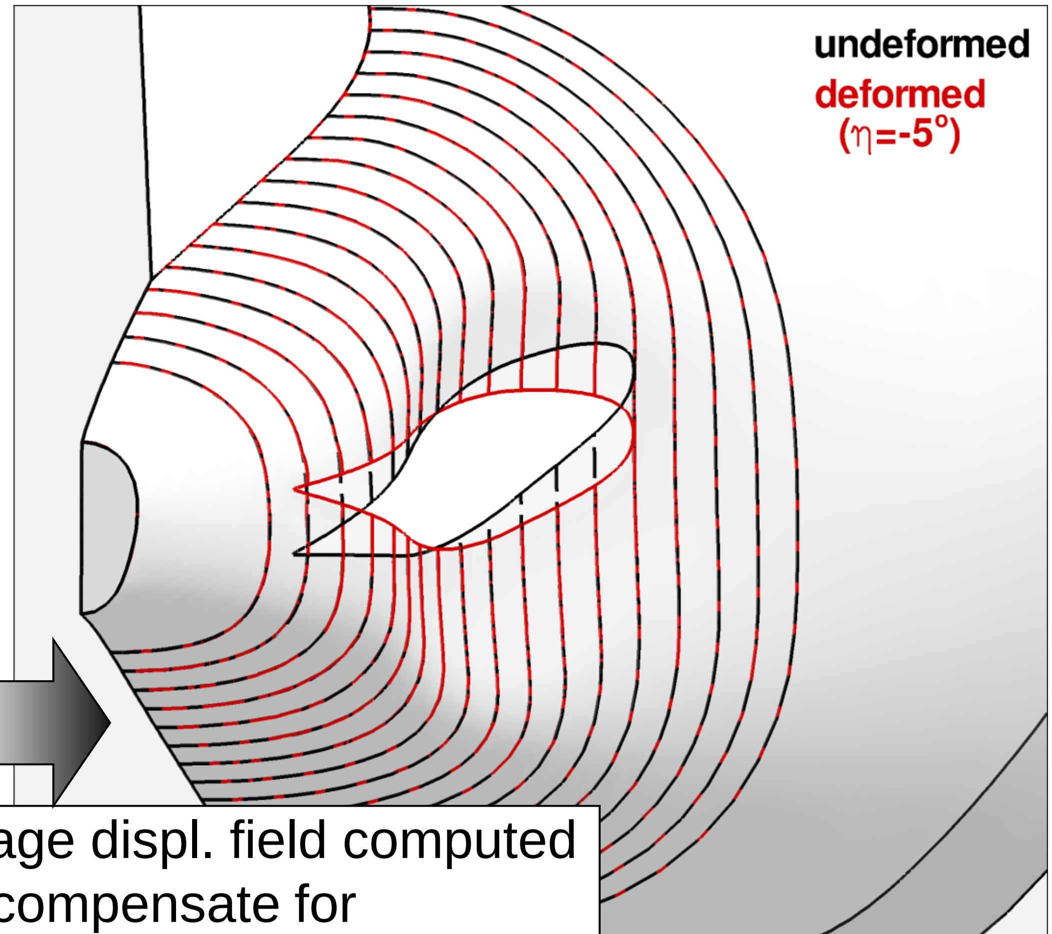
Other Strategy: Special “No-Normal-Movement” BC of FSDeformation



Use of FSDeformations’ No-Normal-Movement BC for Fuselage when Rotating HTPc

BC 1: Displacement field for rigidly rotated HTP



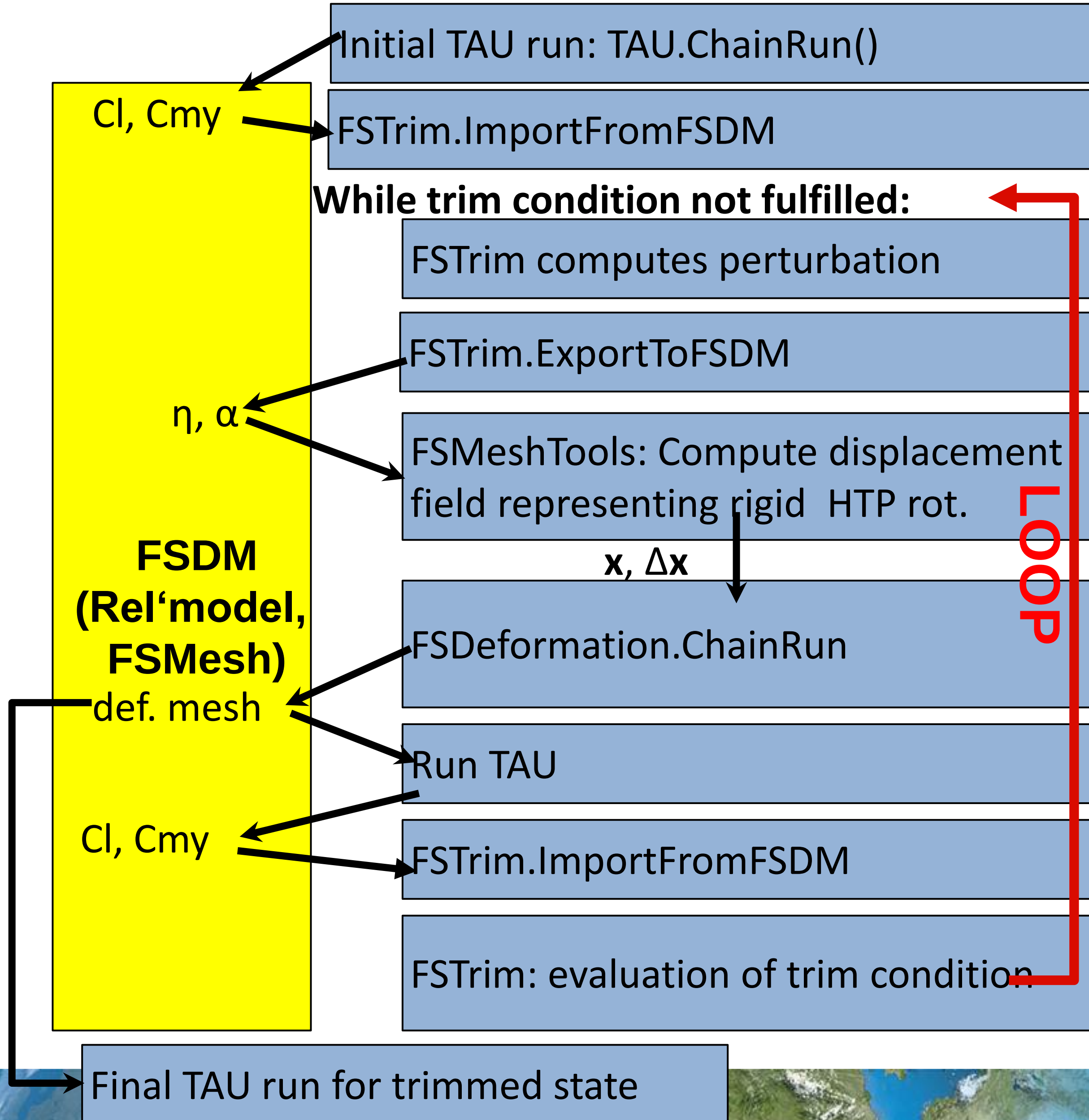


BC 2: Fuselage displ. field computed internally to compensate for geo. violation

Trim Loop Implemented in FlowSimulator

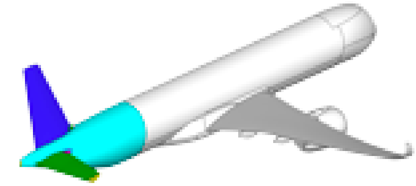
Required Input data:

- Boundary marker of HTP
- Boundary marker of attached fuselage part
- HTP rotation axis (origin, direction)
- Initial HTP setting η
- Initial α
- Target lift, pitch (trimmed state = 0)

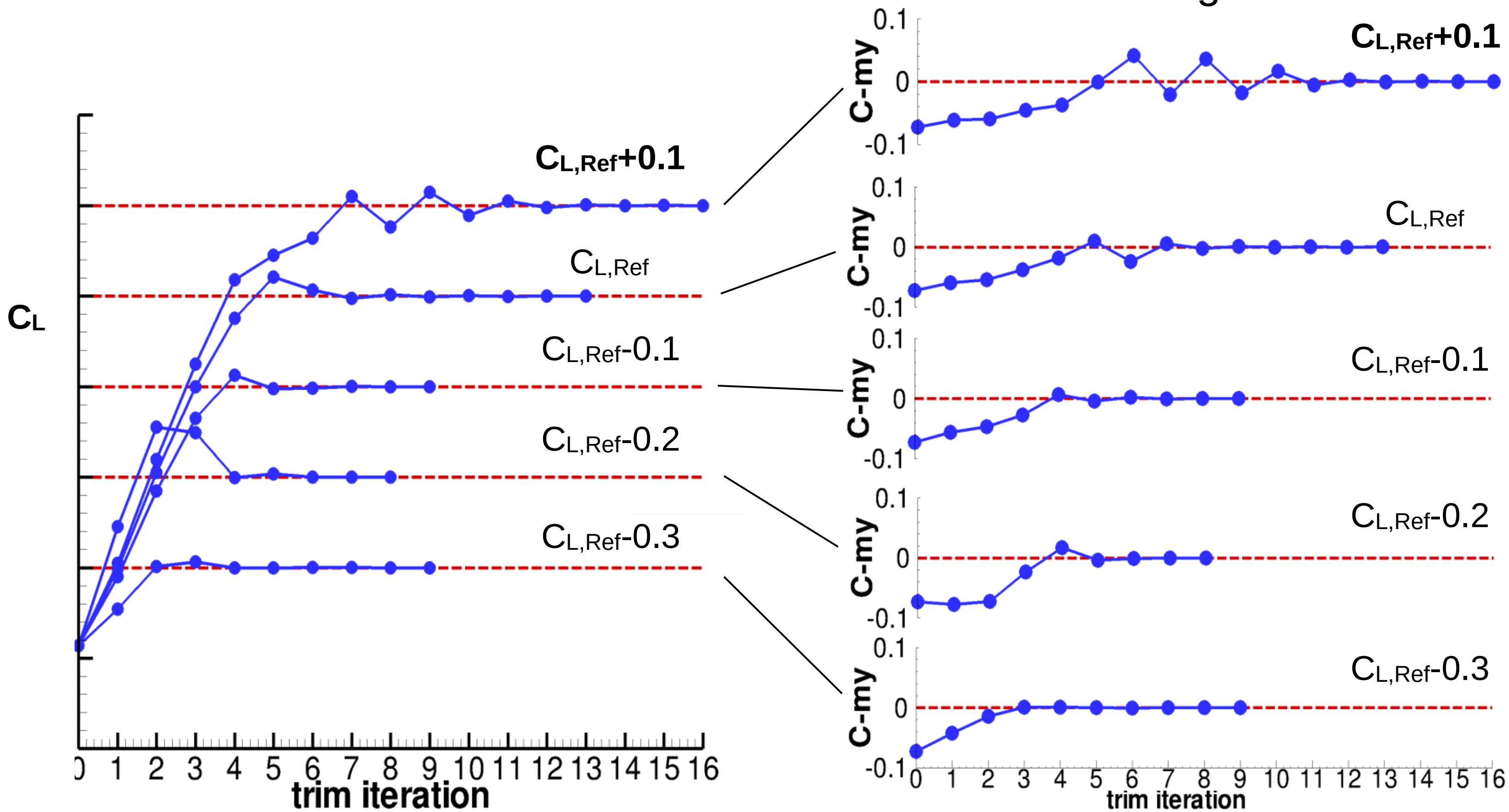


Use Case "Longitudinal Trim"

- FSTau-based simulations
- All computations started from free-stream conditions
- $AoA_{ini}=0°$, $HTP_{ini}=0°$, i.e. untrimmed
- Thrust not considered yet in equation of moments

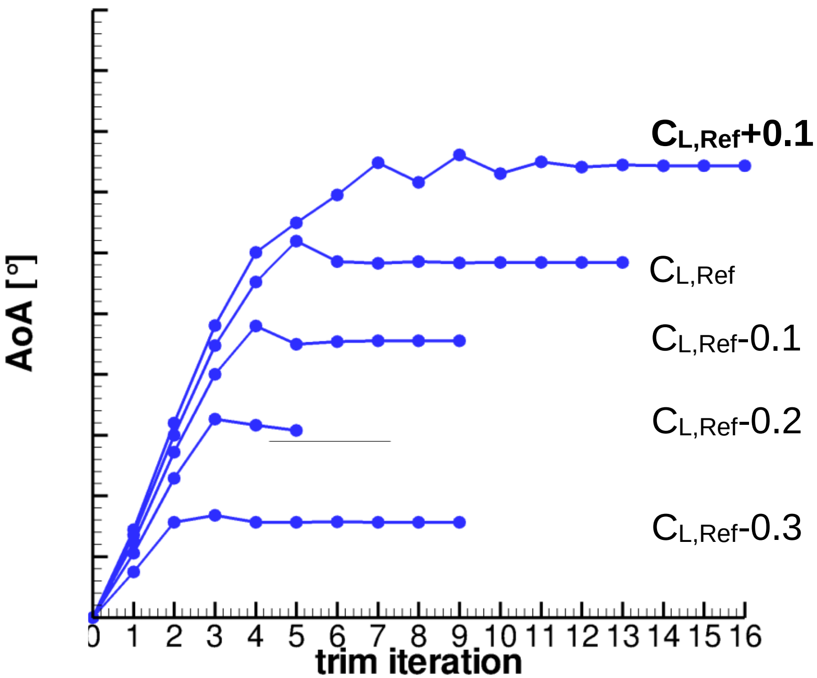


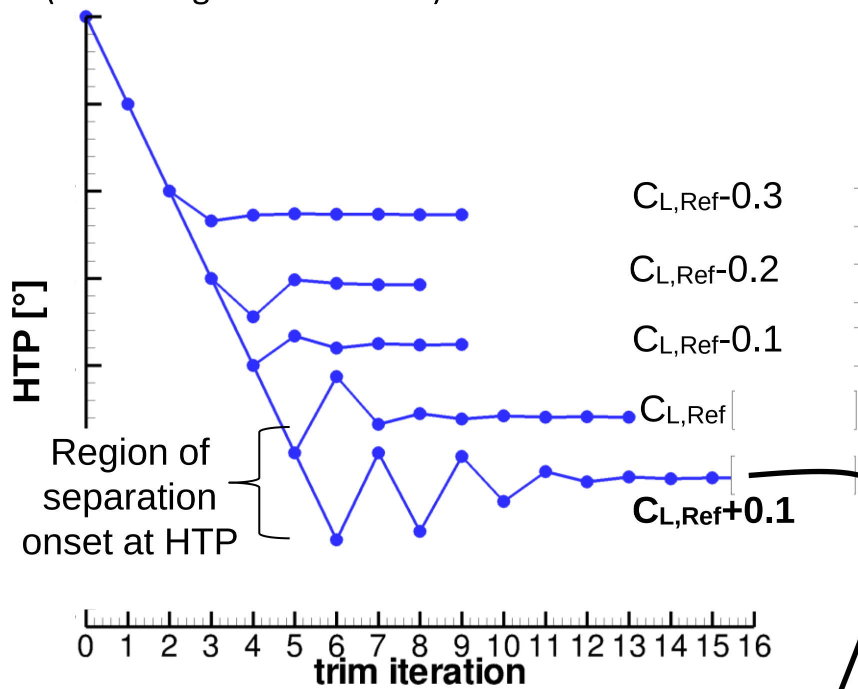
A350-1000 test case: SOLAR mesh, 18m pts., engine BC



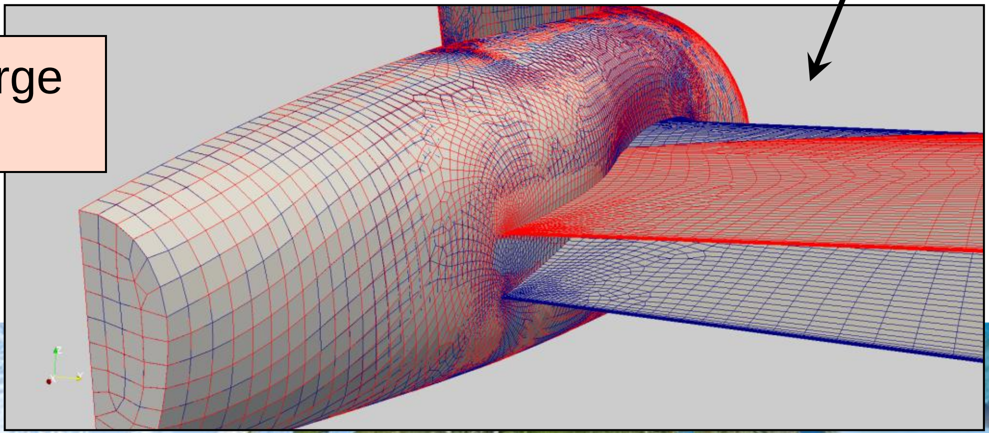
Use Case "Longitudinal Trim"

Remark: HTP 0° indicates the initial setting of the HTP in the CFD mesh (which might be non-zero)



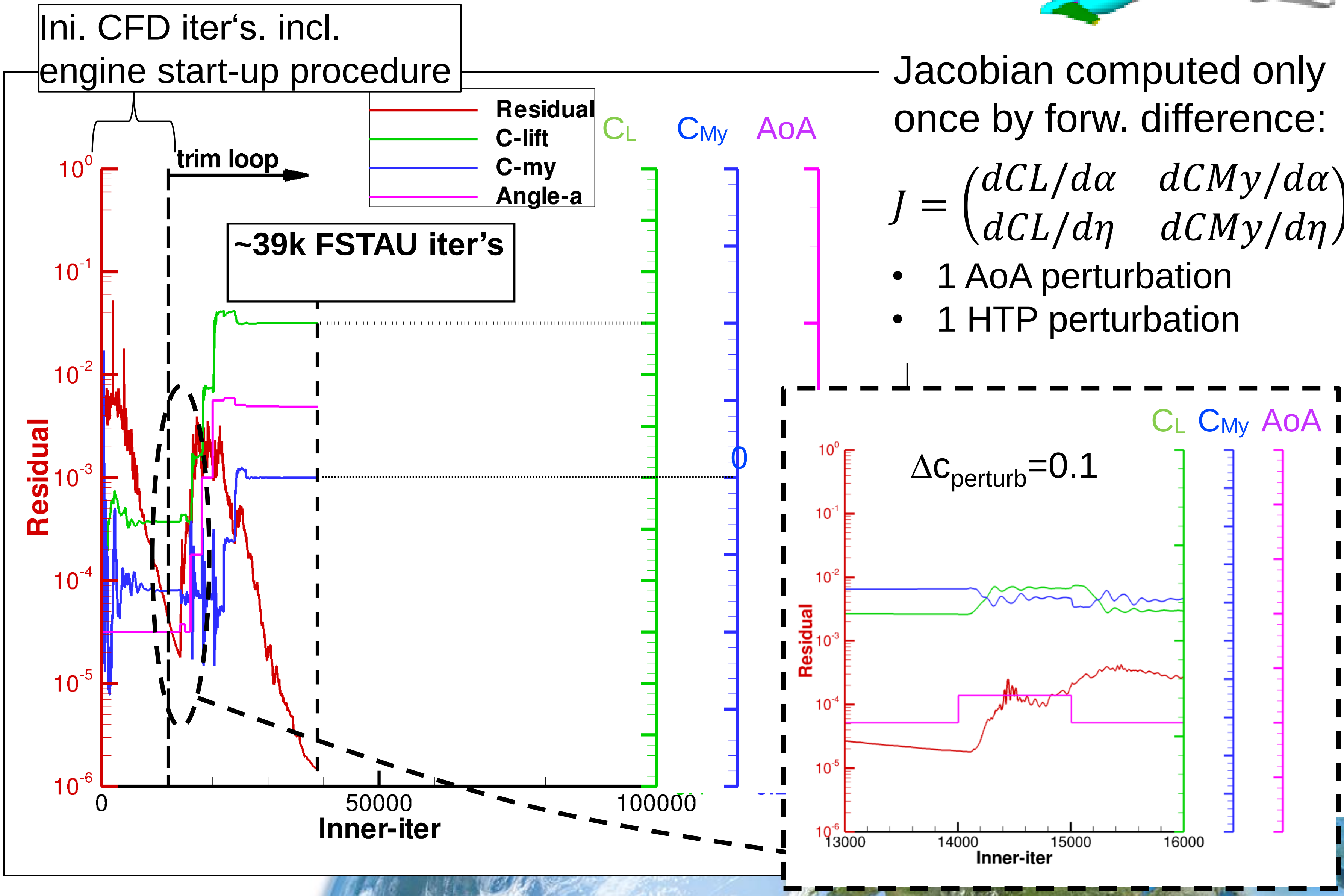


Fuselage shape retained, even for large HTP rotation angles (+1° to -10°)



Use Case "Longitudinal Trim"

A350-1000 test case



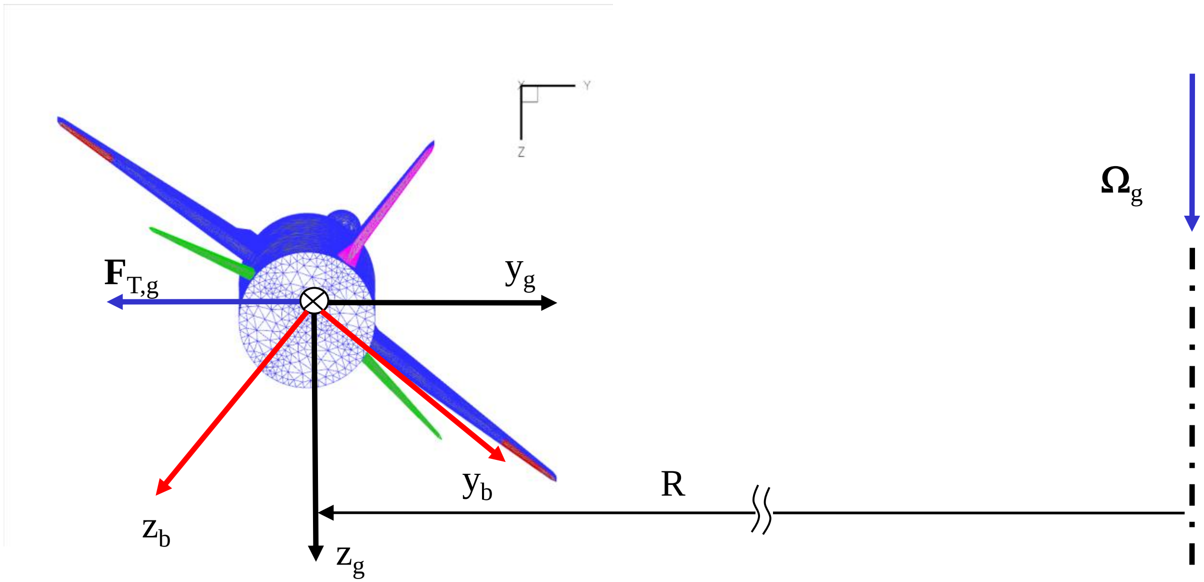
Jacobian computed only once by forw. difference:

$$J = \begin{pmatrix} dCL/d\alpha & dCMy/d\alpha \\ dCL/d\eta & dCMy/d\eta \end{pmatrix}$$

- 1 AoA perturbation
- 1 HTP perturbation

Use Case “Steady Curve Flight”:
Simpler Config., but more Complex Trim Scenario
Involved FS Tools: FSTau + FSTrim + FSDeformation

$$F_{T,g} = m\frac{V_g^2}{R}$$



- We assume that the aircraft is flying a horizontal curve with radius 5000m with constant turn rate (angular velocity Ω_g).
- Flow parameters
  - $Ma_\infty = 0.5$, $p_\infty = 101325\ N / m^2$, $\rho_\infty = 1.29\ kg / m^3$, mass: 9295.44 kg

Use Case “Steady Curve Flight

- Trim-condition / goal function used

Aerodynamic forces and moments

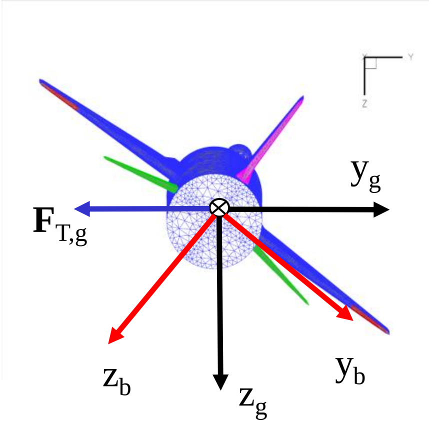
Thrust

Centrifugal force

$$
\begin{aligned}
0 &= F_{x,aero,g} + T_{x,g} \\
0 &= F_{y,aero,g} + T_{y,g} - m\frac{V_g^{\,2}}{R} \\
0 &= F_{z,aero,g} + T_{z,g} + m\,g \\
0 &= M_{x,aero,b} - (I_2 - I_3)q_b r_b \\
0 &= M_{y,aero,b} - (I_3 - I_1)r_b p_b \\
0 &= M_{z,aero,b} - (I_2 - I_3)\,p_b q_b
\end{aligned}
$$

Gravity force

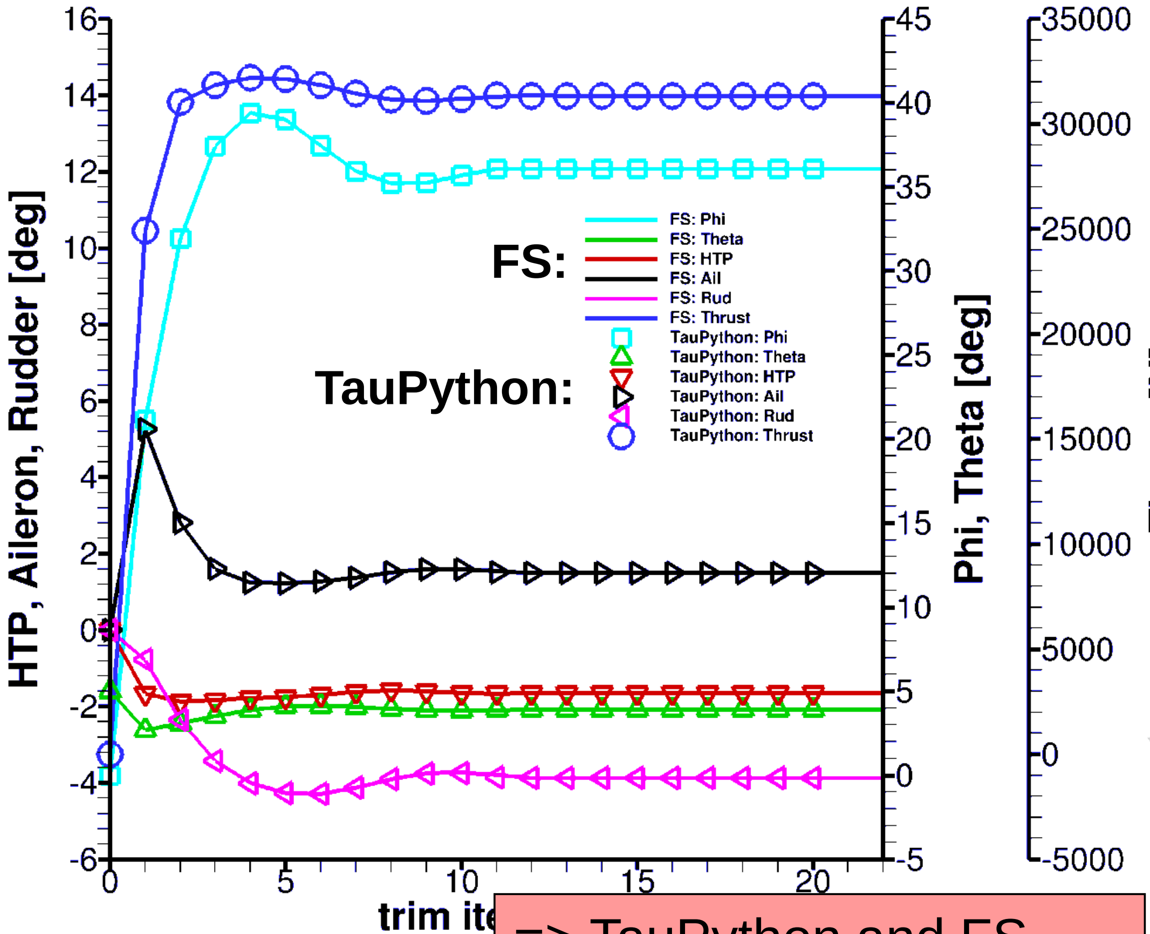
Rate of change of angular momentum



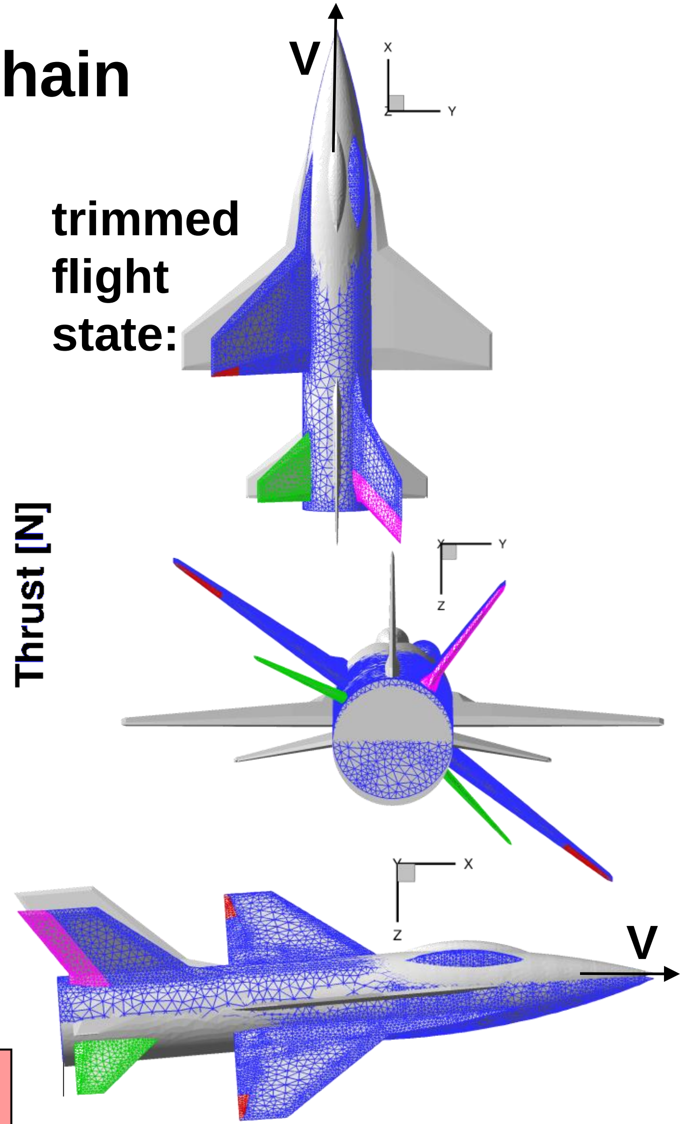
- Control parameters :

Control parameters	Initial values:
Roll angle ϕ	=0°
Pitch angle Θ	=5°
Thrust	=0 N
$\eta_{arileron}$	=0°
η_{rudder}	=0°
η_{HTP}	=0°

Use Case “Steady Curve Flight”: **Comparison with TauPython Tool Chain**



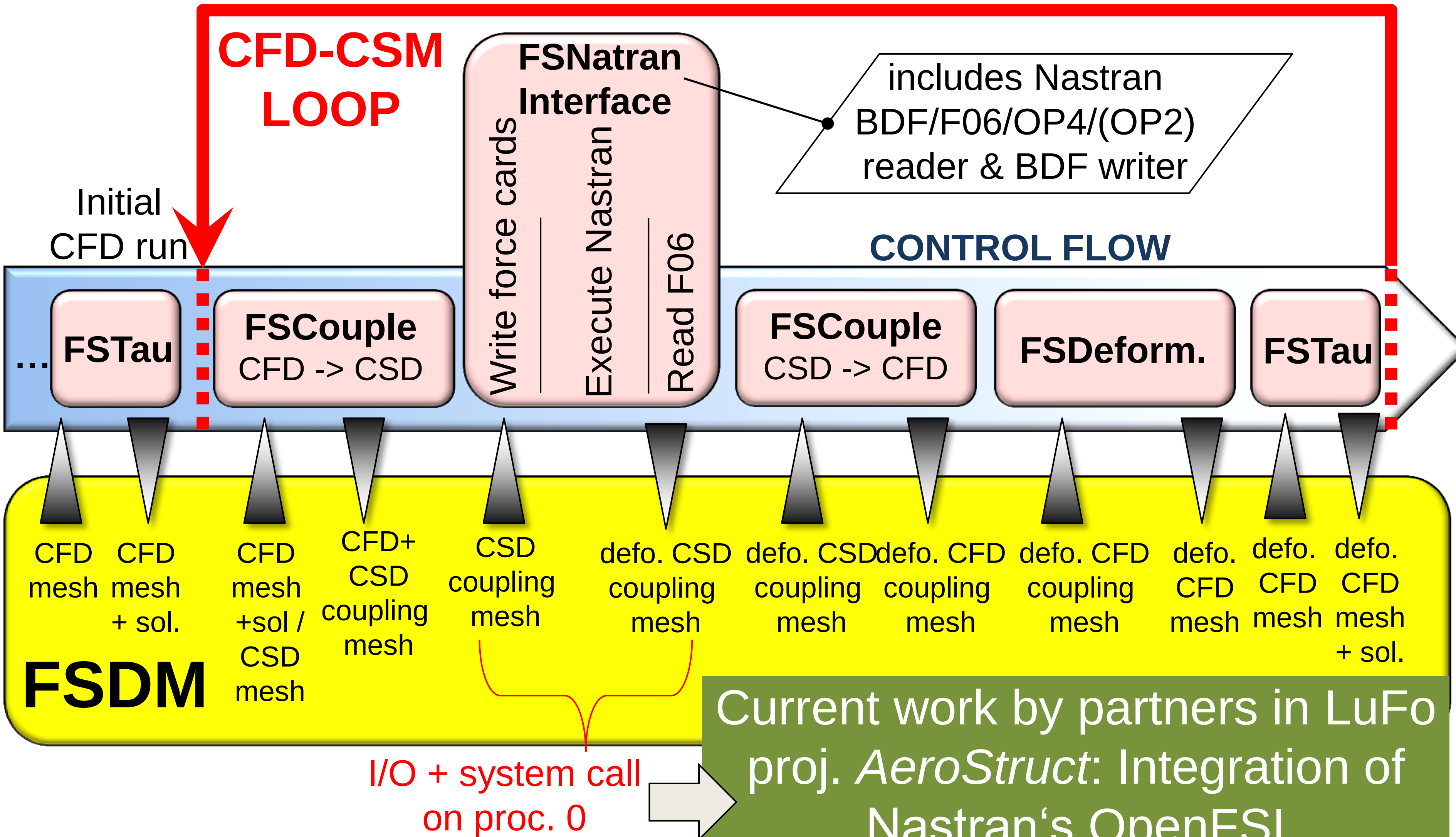
trimmed flight state:



=> TauPython and FS trim procedures yield **identical trim results**

CFD-CSD Coupled Simulations with FlowSimulator

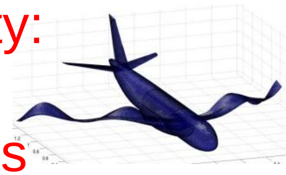
Nastran-based CFD-CSD Coupling with FS

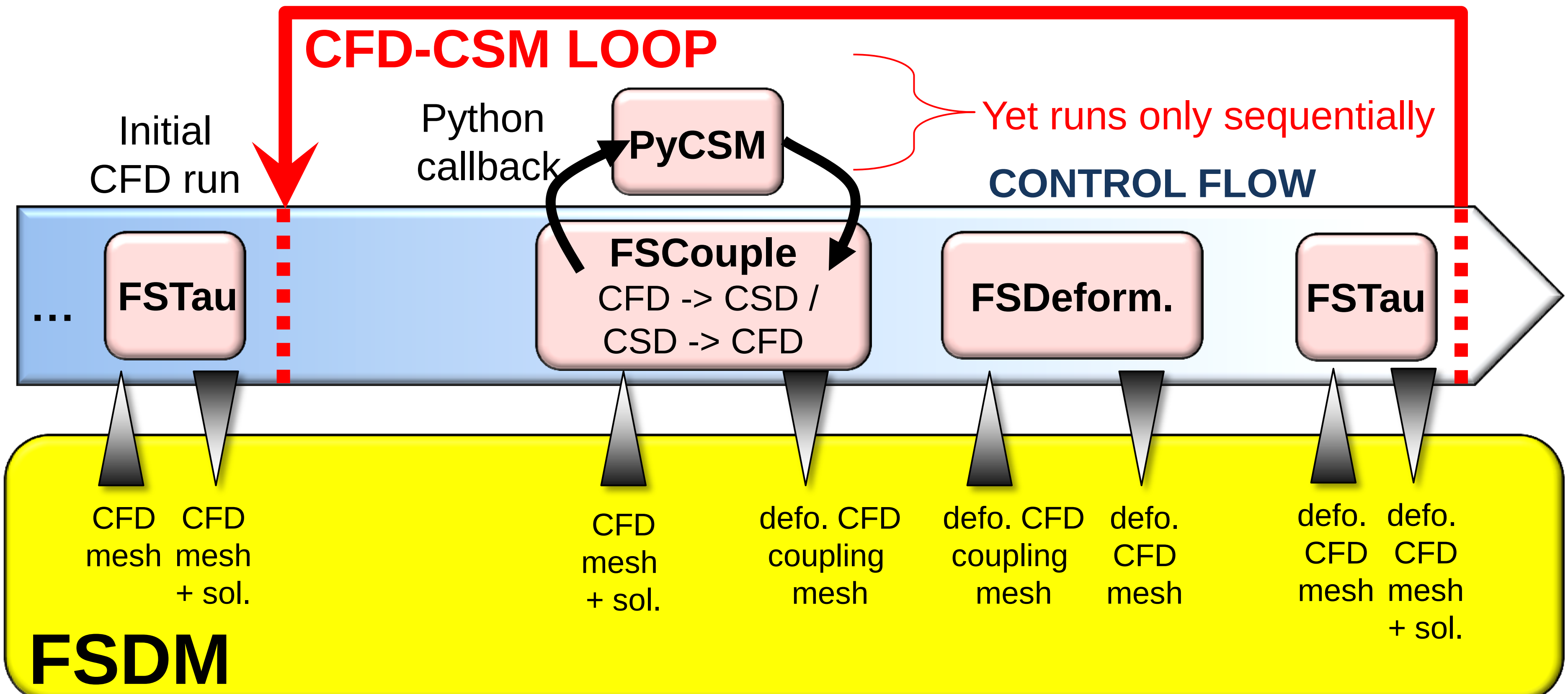


Current work by partners in LuFo proj. *AeroStruct*: Integration of Nastran‘s OpenFSI interface in FS (enables in-memory data exchange)

PyCSM*-based CFD-CSD Coupling with FS

* Python-based modal solver developed by DLR Institute of Aeroelasticity: Includes modal solver + CFD-CSD interpolation methods



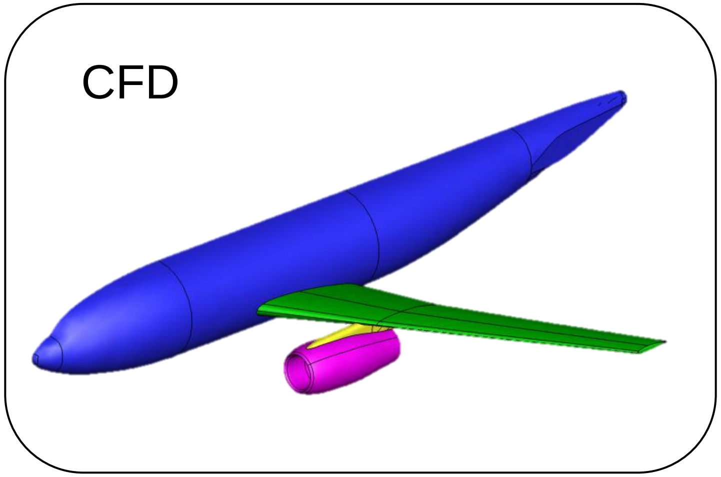


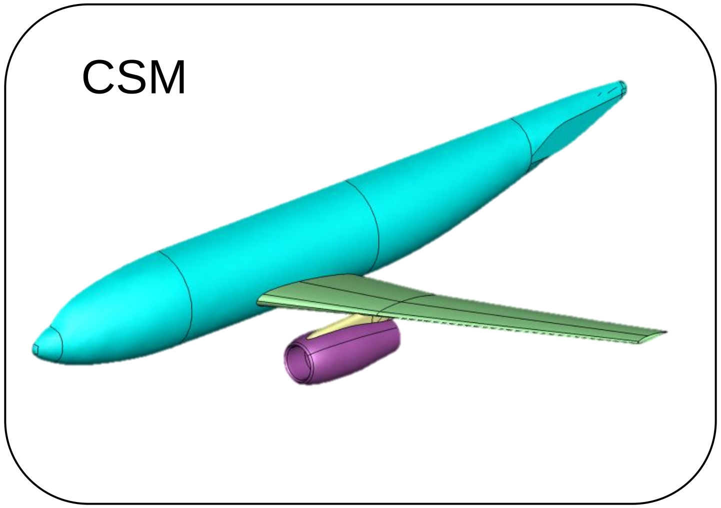
FSCouple

- Originated as a Cassidian development
- Regarded as central component for DLR‘s CFD-CSD coupling chains

Features of FSCouple:

- Definition of pairings (use of FSMeshSelection)



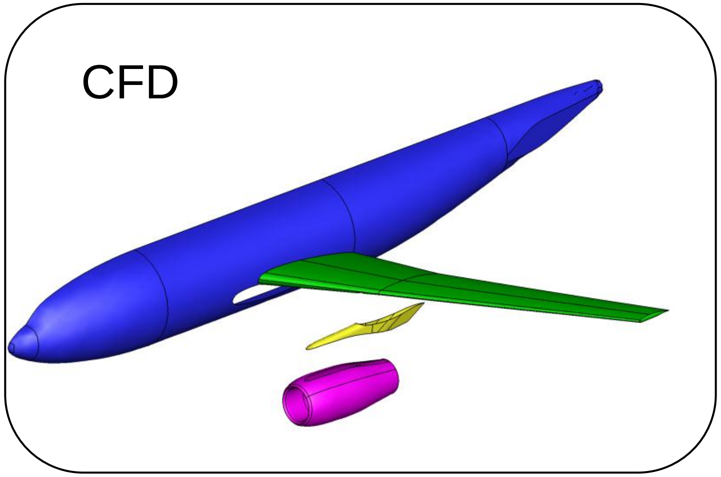


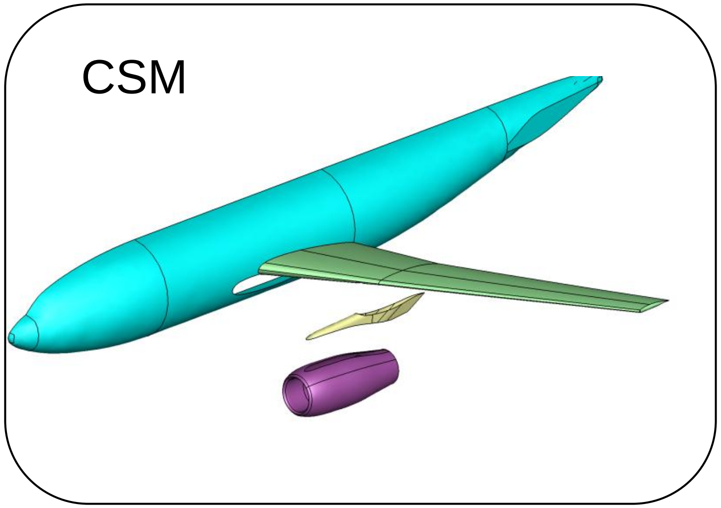
FSCouple

- Originated as a Cassidian development
- Regarded as central component for DLR‘s CFD-CSD coupling chains

Features of FSCouple:

- Definition of pairings (use of FSMeshSelection)



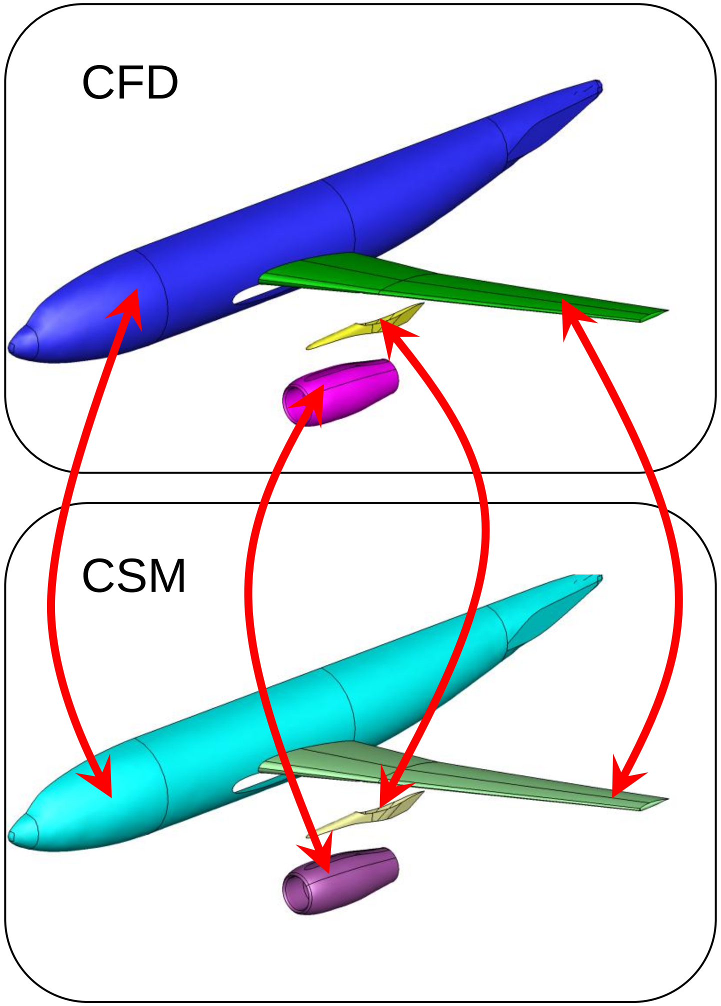


FSCouple

- Originated as a Cassidian development
- Regarded as central component for DLR's CFD-CSD coupling chains

Features of FSCouple:

- Definition of pairings (use of FSMeshSelection)

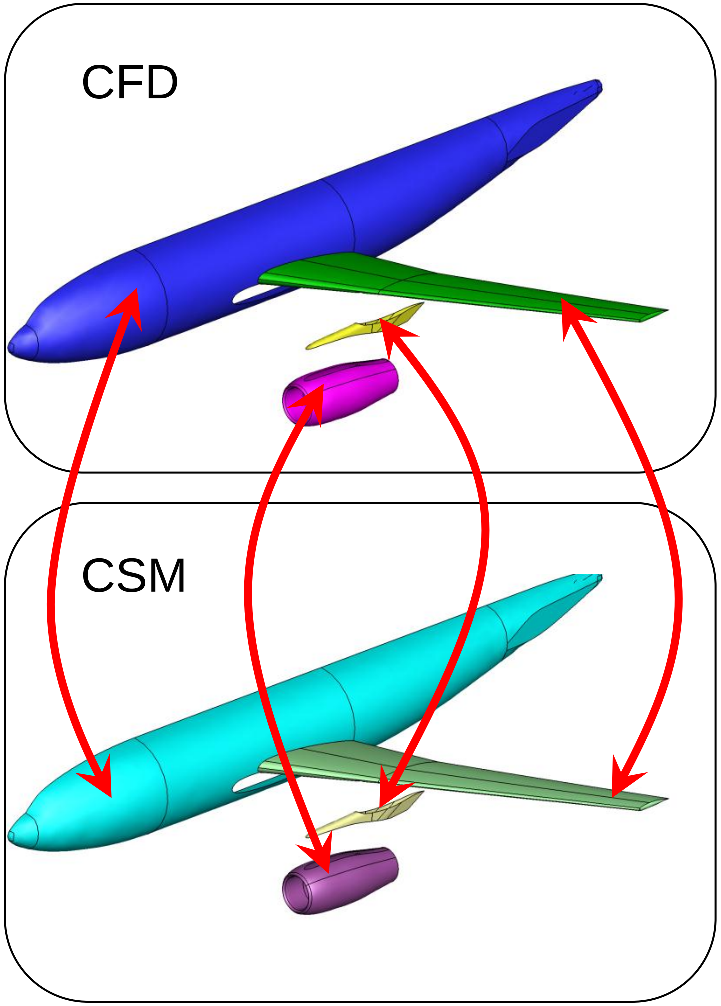


FSCouple

- Originated as a Cassidian development
- Regarded as central component for DLR‘s CFD-CSD coupling chains

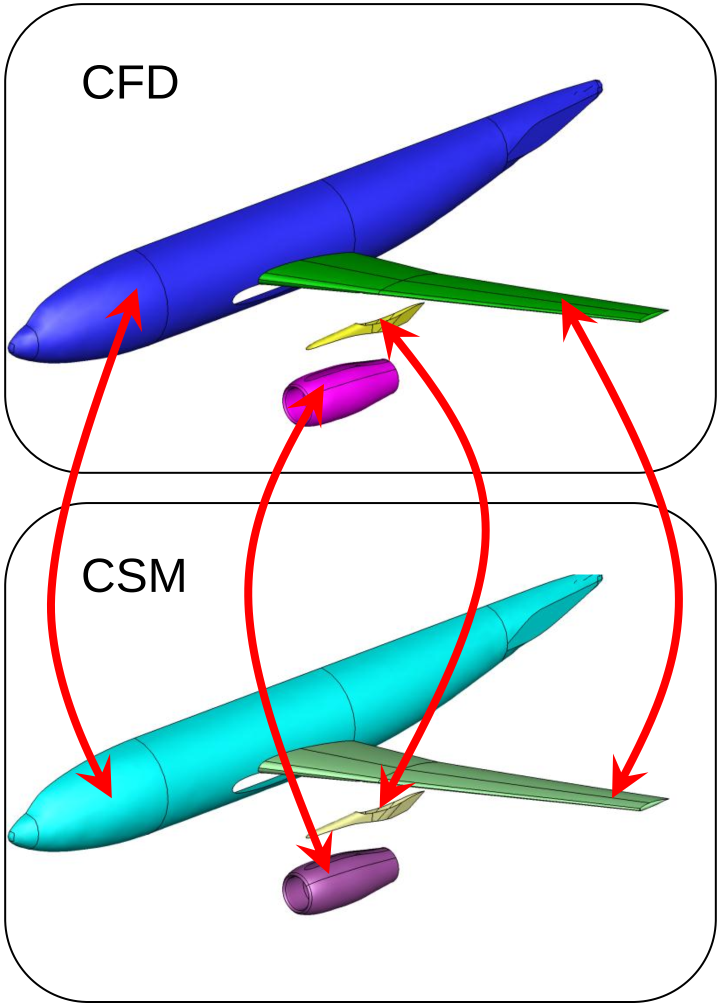
Features of FSCouple:

- Definition of pairings (use of FSMeshSelection)
- Parallel handling of coupling meshes, e.g. repartitioning for improved load balance during interpolation
- Provides interface to interpolation methods (native FSDM methods: FSMeshInterpolation; other external methods via Python callback)



FSCouple

- Originated as a Cassidian development
- Regarded as central component for DLR‘s CFD-CSD coupling chains



Features of FSCouple:

- Definition of pairings (use of FSMeshSelection)
- Parallel handling of coupling meshes, e.g. repartitioning for improved load balance during interpolation
- Provides interface to interpolation methods (native FSDM methods: FSMeshInterpolation; other external methods via Python callback)

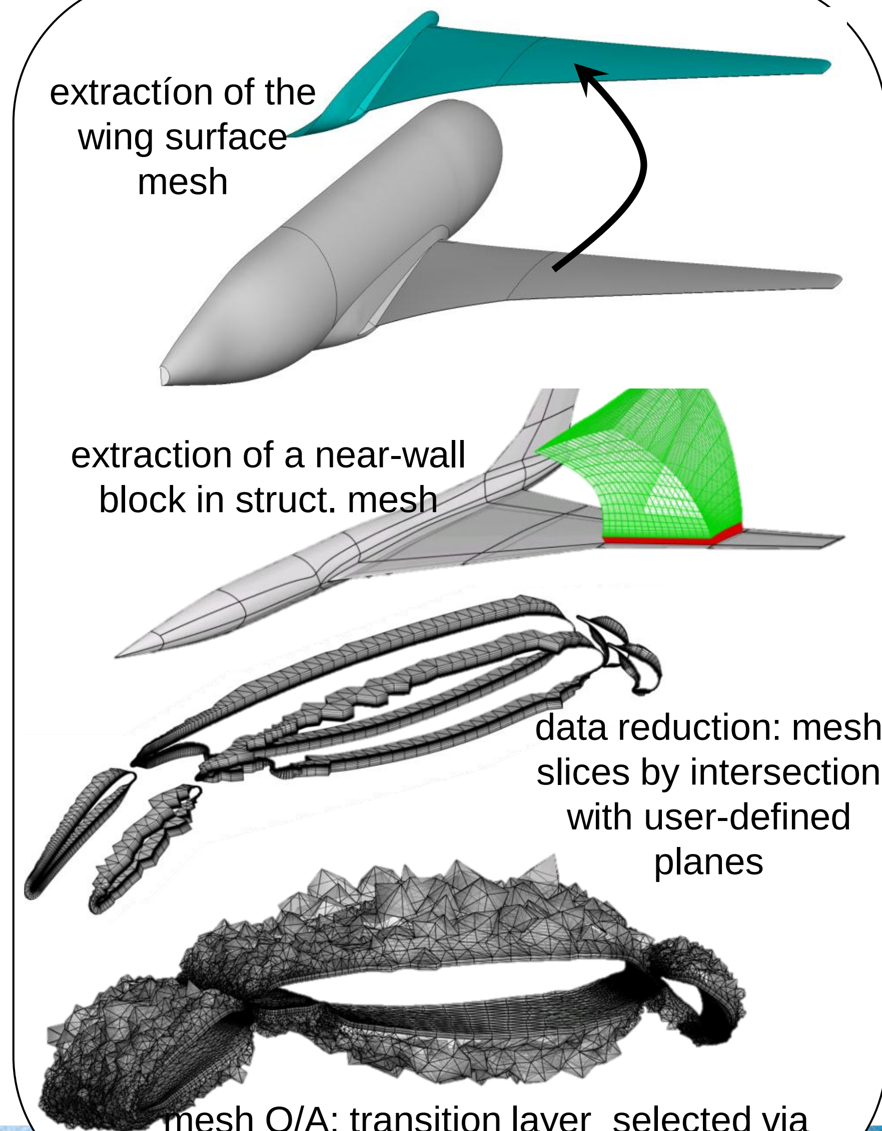
Yet Missing Features:

Inter-pairing blending techniques (planned to be implemented)

FSMeshSelection*

- Efficient in parallel mesh extraction tool
- Applicable to **structured** & **unstructured** meshes
- Allows intuitive boolean operations
- Mandatory to have for CFD-CSD coupling
- Nice to have for data reduction

Example Applications



*part of FSDM (=open source)

FSMeshInterpolation*: Interpolation of Pressure Distrib.

So far available meth's:

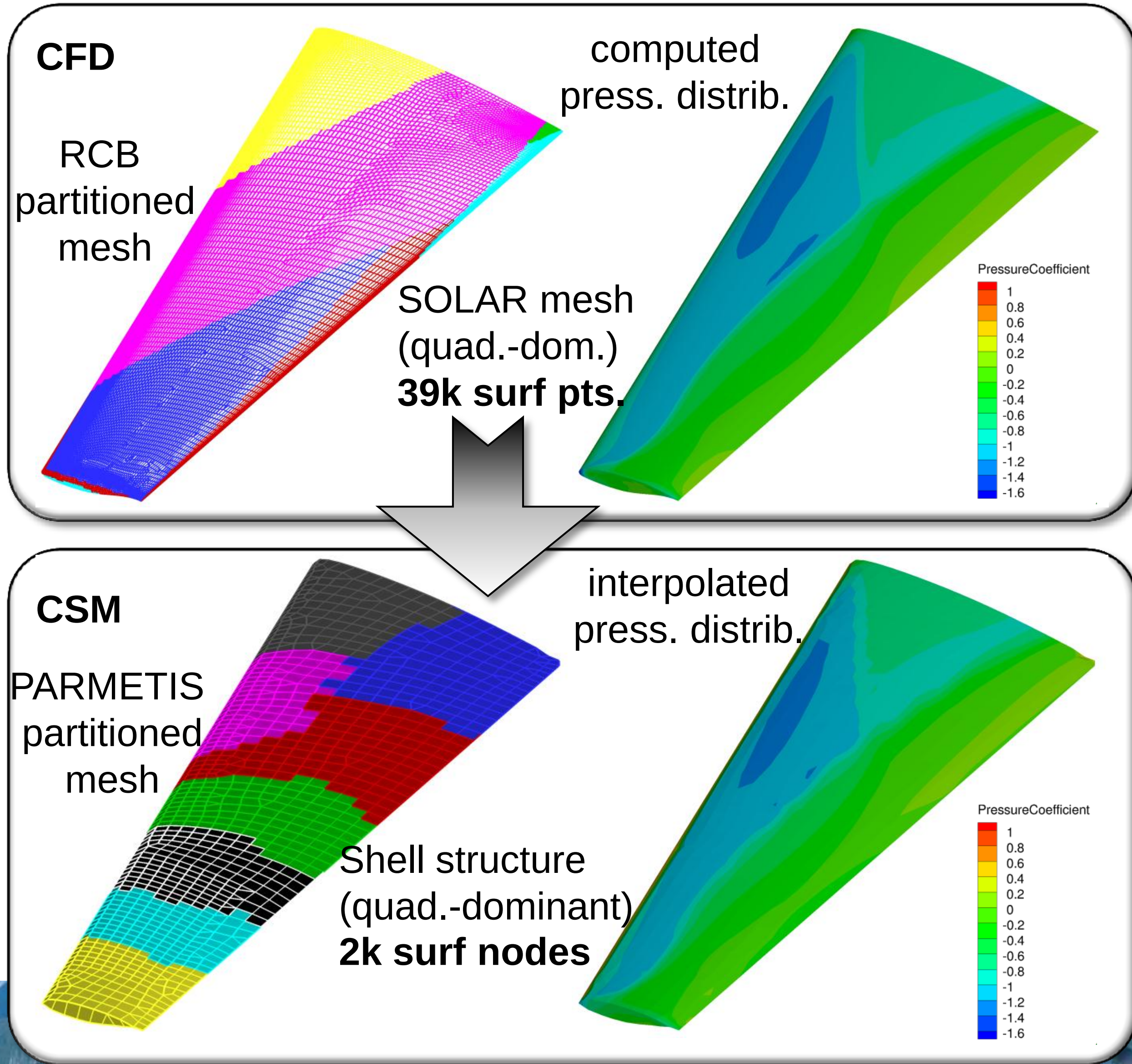
- Volume & surface mesh interpolation
- Neartest neighbour interpolation
- Iso-parametric mapping (FE shape func.)

Planned:

- RBF-based (MLS-type)

*part of FSDM (=open source)

LANN case, FSTAU, M=0.82, Re=7.3m, AoA=2.6°, SAO, 1.7m SOLAR mesh, 8 procs



FSMeshInterpolation*: Interpolation of Pressure Distrib.

So far available meth‘s:

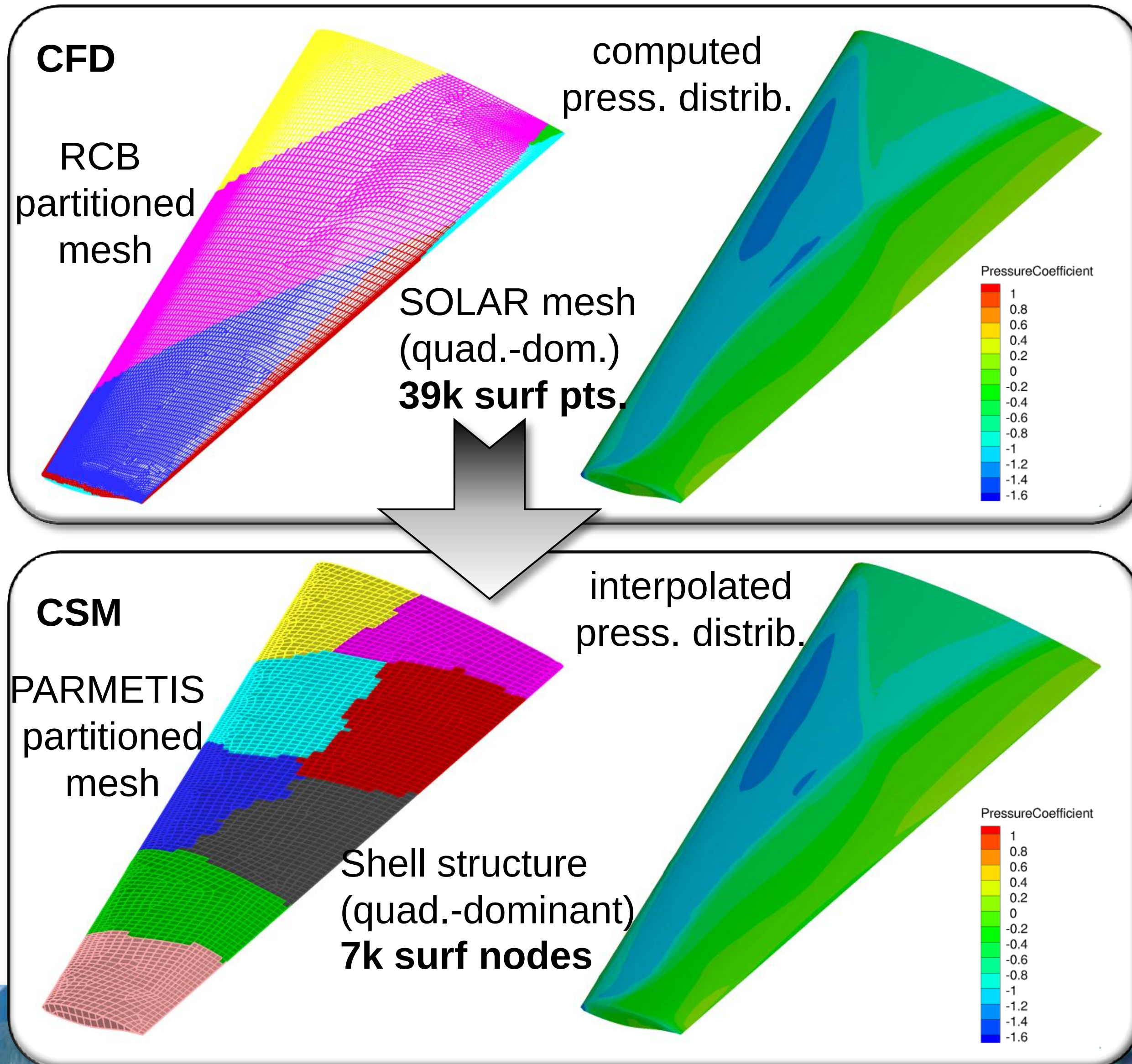
- Volume & surface mesh interpolation
- Neartest neighbour interpolation
- Iso-parametric mapping (FE shape func.)

Planned:

- RBF-based (MLS-type)

*part of FSDM (=open source)

LANN case, FSTAU, M=0.82, Re=7.3m, AoA=2.6°, SAO, 1.7m SOLAR mesh, 8 procs



FSMeshInterpolation*: Interpolation of Pressure Distrib.

So far available meth's:

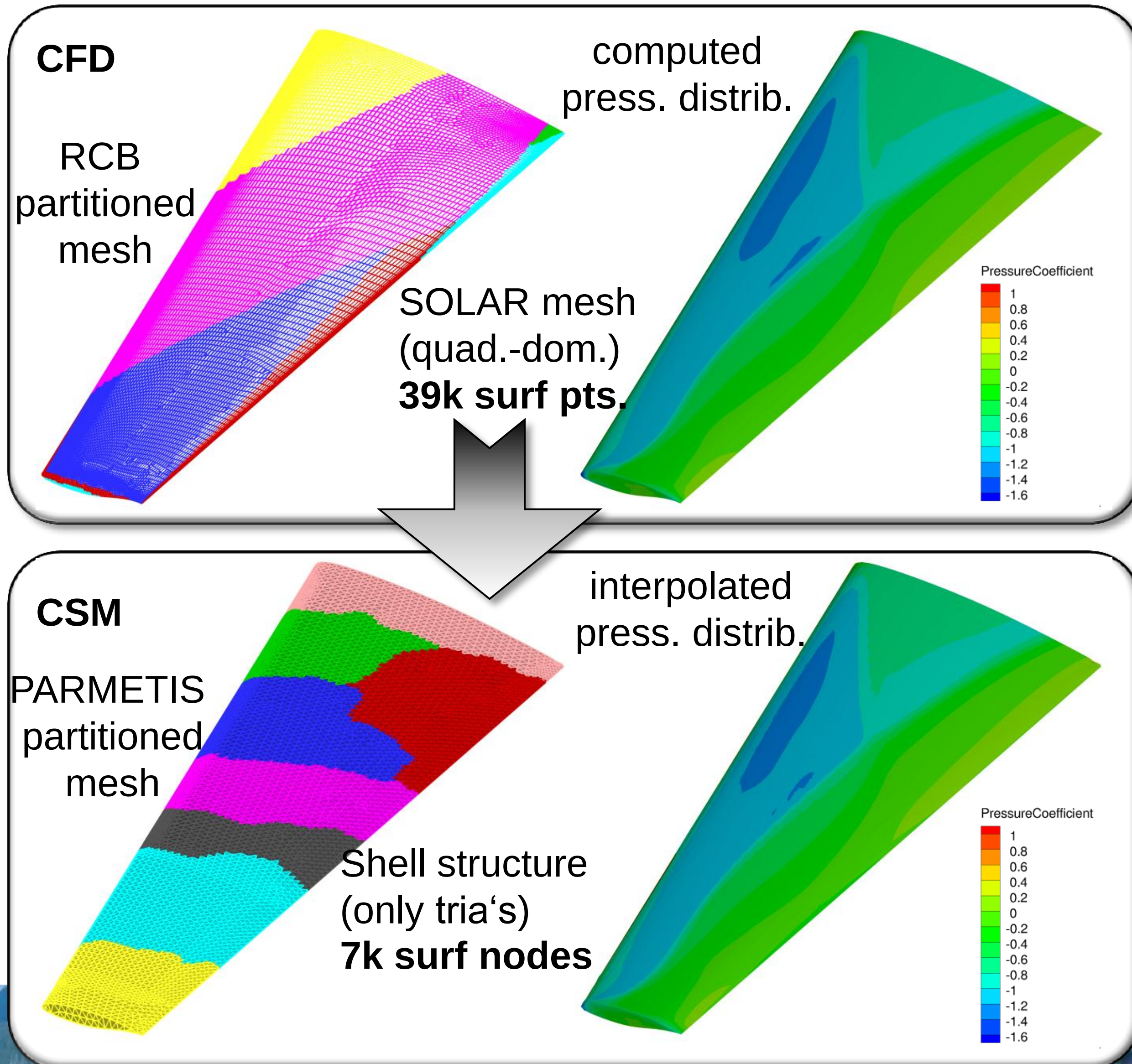
- Volume & surface mesh interpolation
- Neartest neighbour interpolation
- Iso-parametric mapping (FE shape func.)

Planned:

- RBF-based (MLS-type)

*part of FSDM (=open source)

LANN case, FSTAU, M=0.82, Re=7.3m, AoA=2.6°, SAO, 1.7m SOLAR mesh, 8 procs



DLR

FSMeshInterpolation*: Interpolation of Pressure Distrib.

So far available meth's:

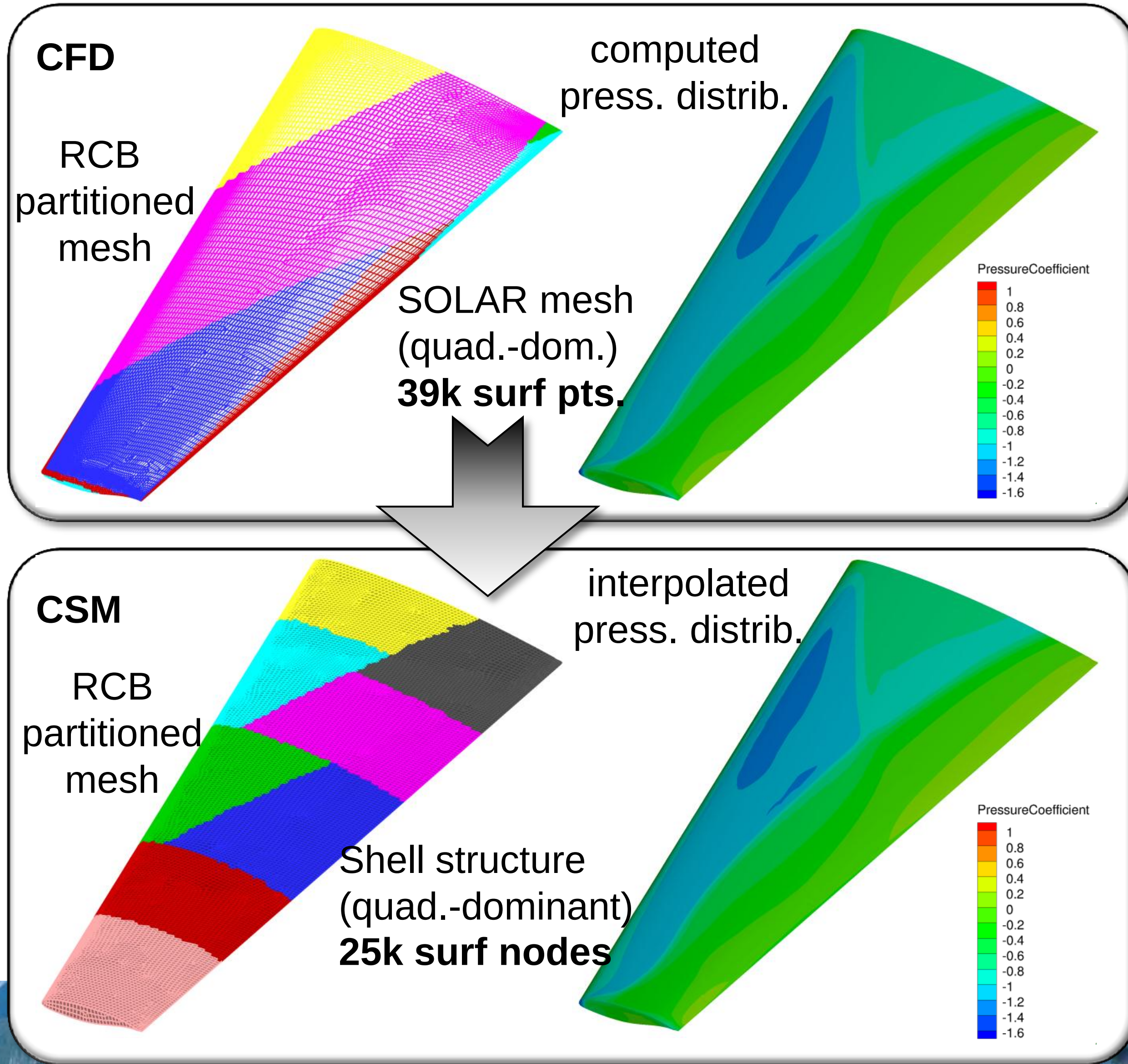
- Volume & surface mesh interpolation
- Neartest neighbour interpolation
- Iso-parametric mapping (FE shape func.)

Planned:

- RBF-based (MLS-type)

*part of FSDM (=open source)

LANN case, FSTAU, M=0.82, Re=7.3m, AoA=2.6°, SAO, 1.7m SOLAR mesh, 8 procs



FSMeshInterpolation*: Interpolation of Displacements

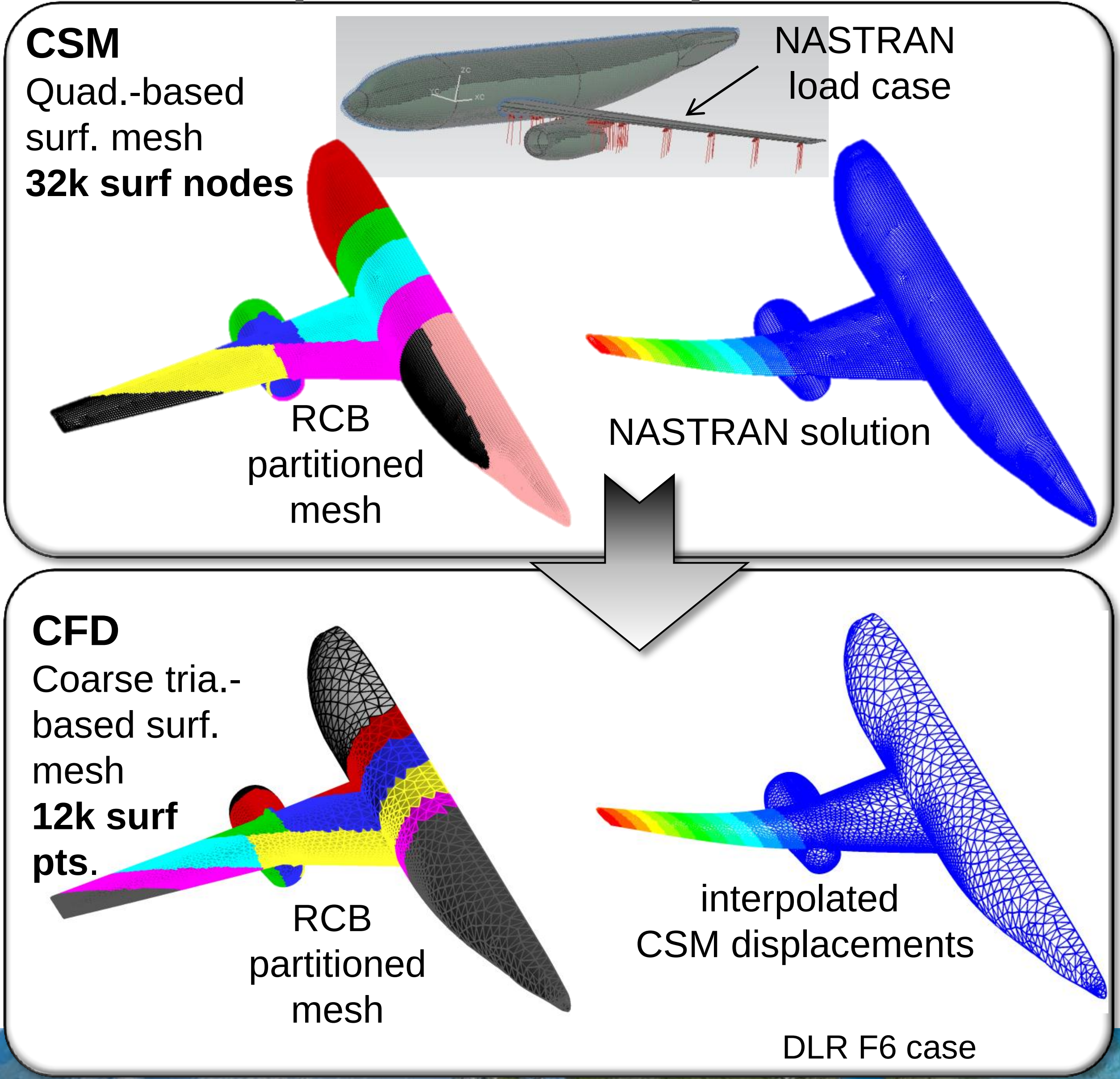
So far available meth‘s:

- Volume & surface mesh interpolation
- Neartest neighbour interpolation
- Iso-parametric mapping (FE shape func.)

Planned:

- RBF-based (MLS-type)

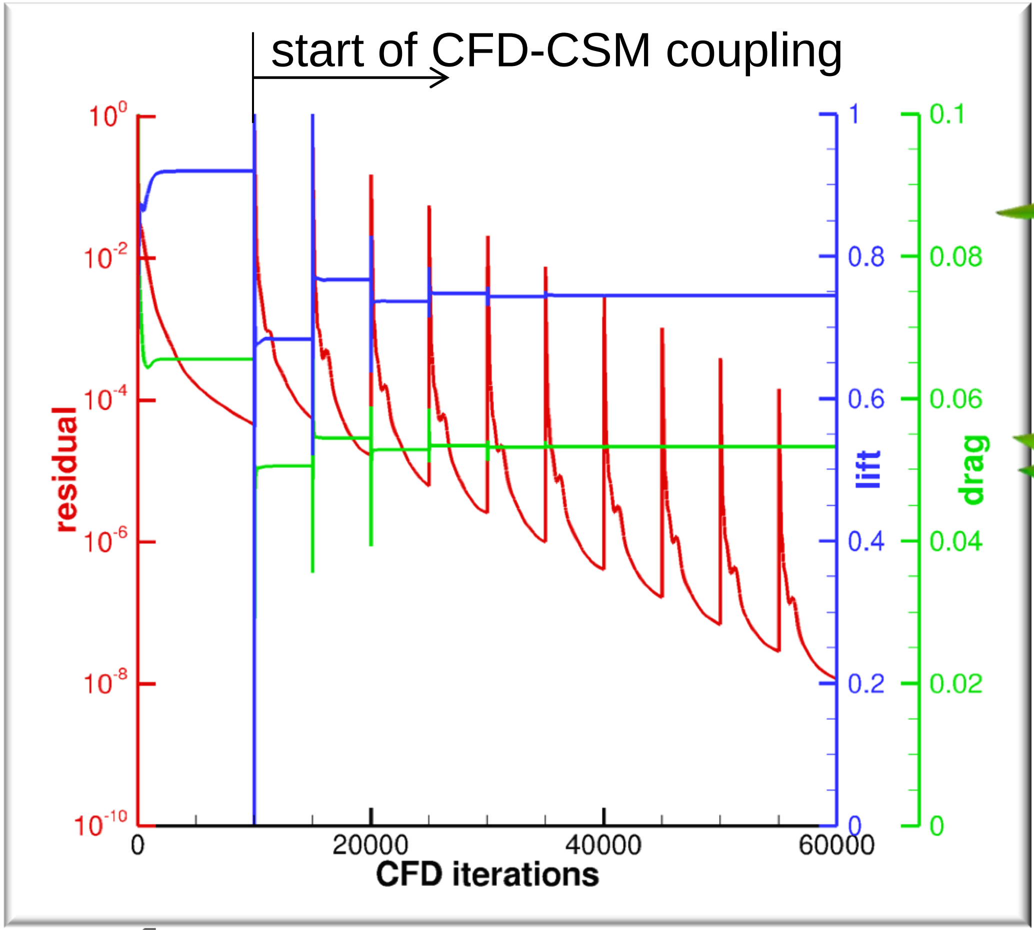
*part of FSDM (=open source)

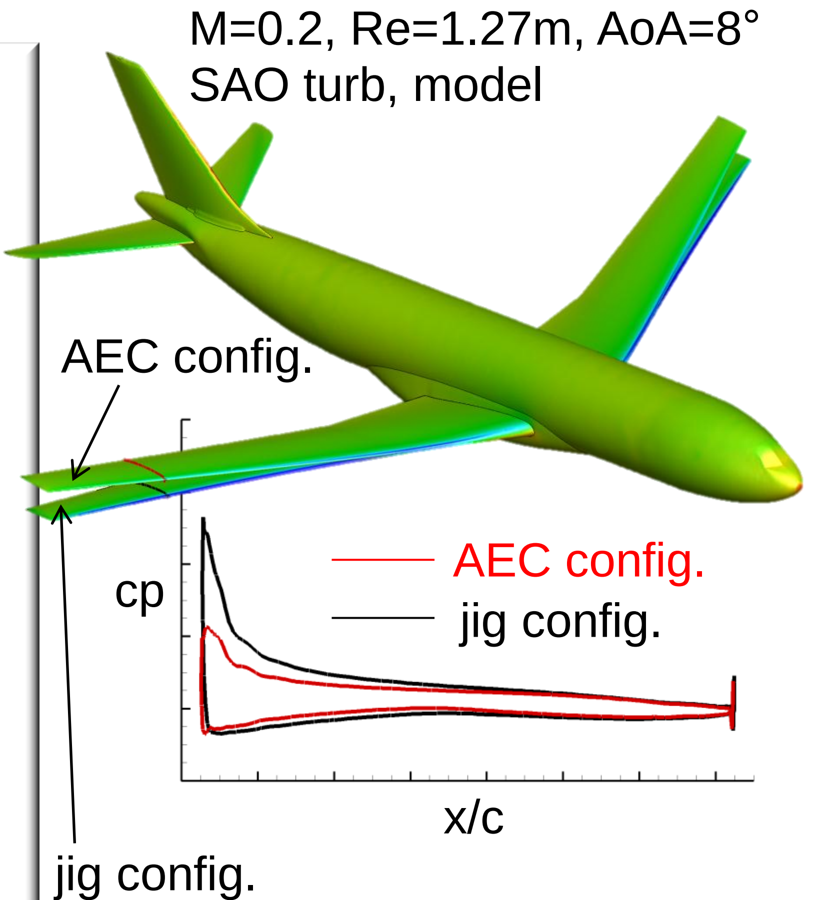


Use case „F12“: Static Aeroelastic Simulation

FSTau + FSCouple + PyCSM + FSDeformation

20 CSD modes used, PyCSM on proc 0



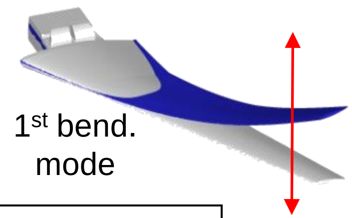


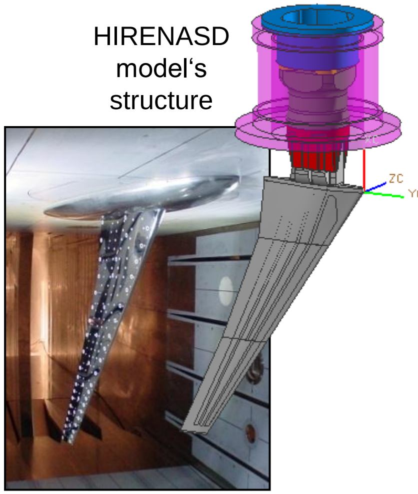
Use case „HIRENASD“: Dynamic Aeroelastic Simulation

FSTau + FSCouple + PyCSM + FSDeformation

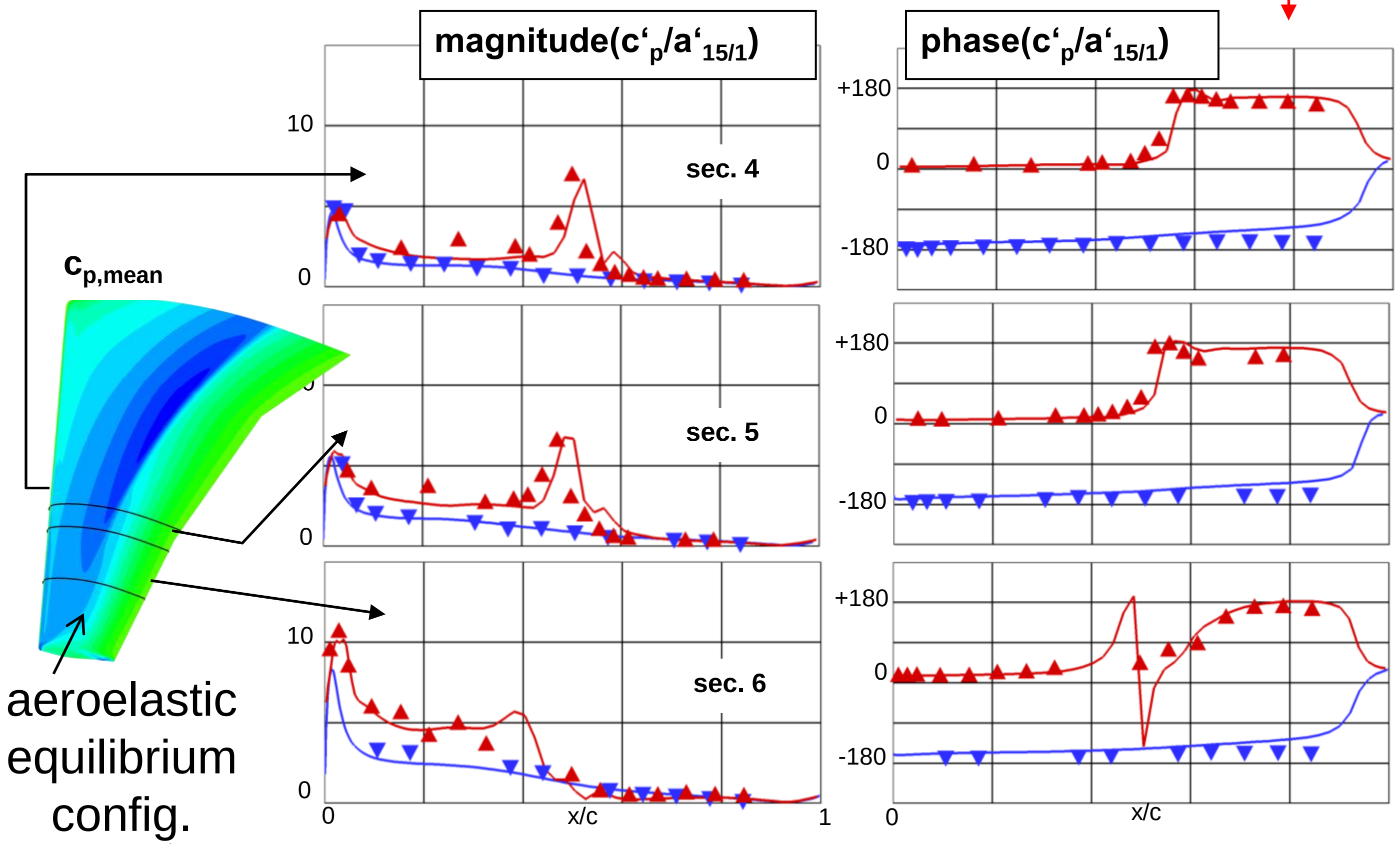
Test 143:

- M=0.8, Re=7m, α=1.5, f_{exc}=26.92Hz
- Excitation of 1st bending mode



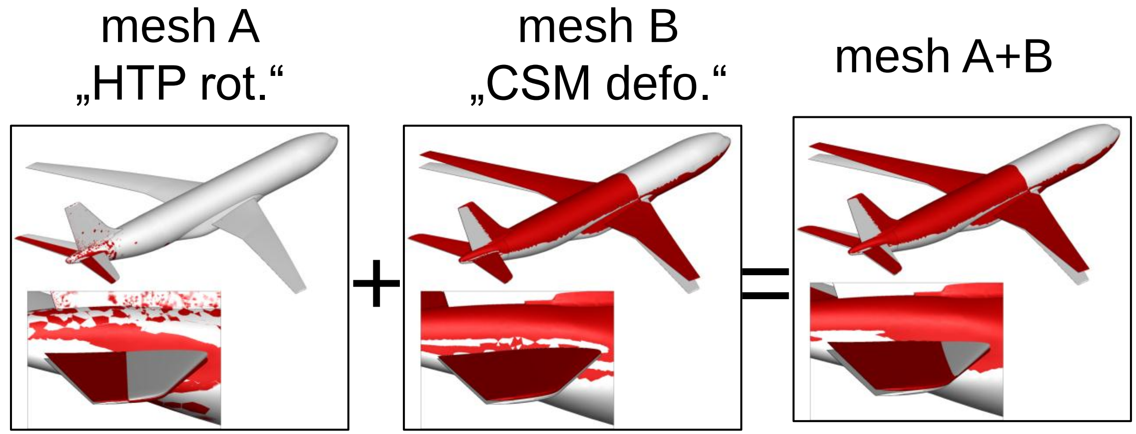


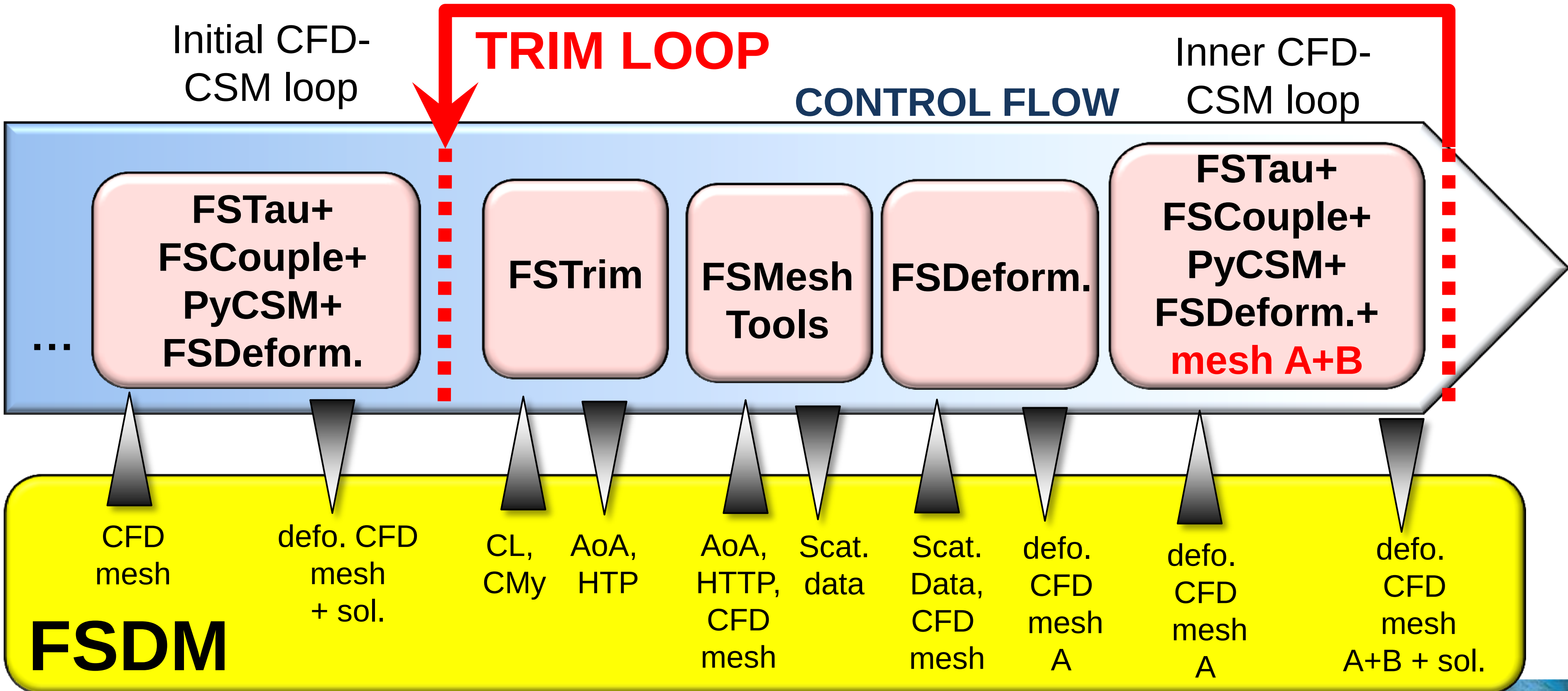
photograph of HIRENASD model in ETW's test section



Results by courtesy of DLR Institute for Aeroelasticity

Current Work: Combination of Trim and CFD-CSD Coupling

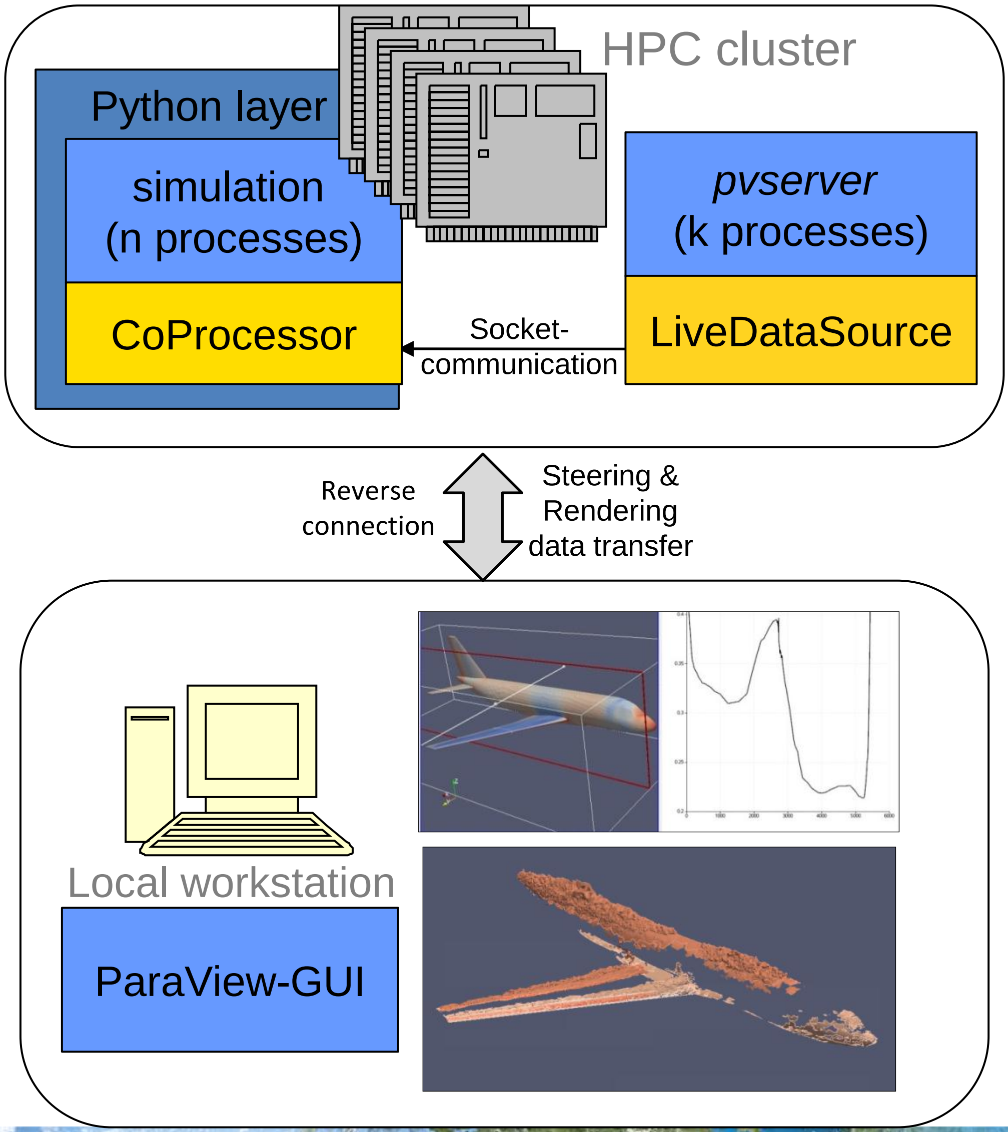




Co-processing-based Visualisation

Benefits:

- Online monitoring
- Data reduction
- In parallel with sim. w/o file I/O
- Parallel graphical data proc.

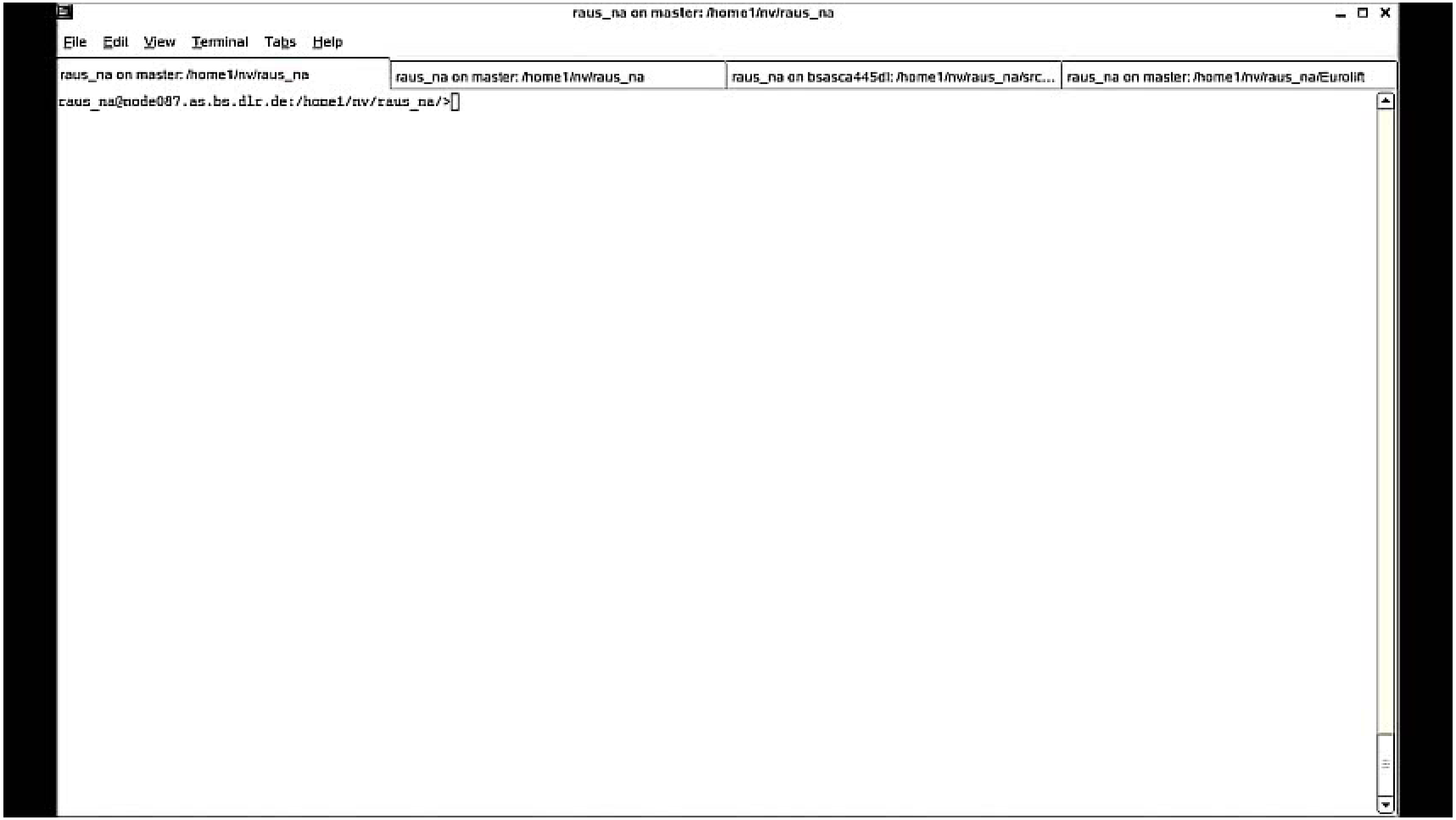


Implementation:

- Based on ParaView (version 3.9 + small own modifications)
- Uses ParaView-CoProcessing lib for Python-based integration in simulation
- FS interface available and applied at Airbus

DLR

Co-processing-based Visualisation: Demonstration



Summary

- Potential of FS demonstrated for multidisciplinary parallel analysis
- DLR strives to use FS for all multidisciplinary simulations in the near future
- DLR takes over maintenance of FSDeformation
- DLR will further develop FSTrim, FSDeformation, FSCouple, FSDM and will contribute FS6DoF
- With ongoing FS development more synergy effects of ONERA and DLR contributations can be expected

Examples of Currently Running Projects Involving DLR with Extense Usage and Development of FlowSImulator

- DLR project DIGITAL X
- German Aerospace Research Programme (LuFo) project AeroStruct
- CFD4Loads project (initiated by Airbus)

...

End